

A New Simulation-based Robust Optimal Design of FOPID Controller for Five Bar Linkage Robot in a Cyber-physical System

Amir Pamianifard

Department of Electrical Engineering, Faculty of Engineering, Chulalongkorn University, Bangkok 10330, Thailand. <https://orcid.org/0000-0002-0760-2149>

Lunchakorn Wuttisittikulij (✉ Lunchakorn.W@chula.ac.th)

Department of Electrical Engineering, Faculty of Engineering, Chulalongkorn University, Bangkok 10330, Thailand.

Ali Zemouche

University of Lorraine, CRAN UMR CNRS 7039, 54400 Cosnes et Romain, France.

Gridsada Phanomchoeng

Department of Mechanical Engineering, Faculty of Engineering, Chulalongkorn University, Bangkok 10330, Thailand

Research article

Keywords: Robust Optimal Control, Cyber-Physical System, Gaussian Process, Particle Swarm Optimization, Robotics Manipulators, Multiple Setpoint

Posted Date: August 5th, 2020

DOI: <https://doi.org/10.21203/rs.3.rs-52777/v1>

License:  This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

A New Simulation-Based Robust Optimal Design of FOPID Controller for Five Bar Linkage Robot in a Cyber-Physical System

Amir Parnianifard¹, Lunchakorn Wuttisittikulij¹, Ali Zemouche², Gridsada Phanomchoeng³

1. Department of Electrical Engineering, Faculty of Engineering, Chulalongkorn University, Bangkok 10330, Thailand.
2. University of Lorraine, CRAN UMR CNRS 7039, 54400 Cosnes et Romain, France.
3. Department of Mechanical Engineering, Faculty of Engineering, Chulalongkorn University, Bangkok 10330, Thailand

* Corresponding author: Lunchakorn Wuttisittikulij Email: Lunchakorn.W@chula.ac.th

ABSTRACT:

This paper aims to further increase the reliability of optimal results by setting the simulation conditions to be as close as possible to the real or actual operation to create a Cyber-Physical System (CPS) view for the installation of the Fractional-Order PID (FOPID) controller. For this purpose, we consider two different sources of variability in such a CPS control model. The first source refers to changeability of a target of control model (multiple setpoints) because of environmental noise factors and the second source refers to an anomaly in sensors that is raised in a feedback loop. We develop a new approach to optimize two objective functions under uncertainty including signal energy control and response error control while obtaining the robustness among the source of variability with the lowest computational cost. A new hybrid surrogate-metaheuristic approach is developed using Particle Swarm Optimization (PSO) to update the Gaussian Process (GP) surrogate for a sequential improvement of the robust optimal result. The application of efficient global optimization is extended to estimate surrogate prediction error with less computational cost using a jackknife leave-one-out estimator. This paper examines the challenges of such a robust multi-objective optimization for FOPID control of a five-bar linkage robot manipulator. The results show the applicability and effectiveness of our proposed method in obtaining robustness and reliability in a CPS control system by tackling required computational efforts.

KEYWORDS: *Robust Optimal Control, Cyber-Physical System, Gaussian Process, Particle Swarm Optimization, Robotics Manipulators, Multiple Setpoint.*

1. Introduction

Nowadays, developing processes in the engineering world are strongly associated with computer simulations. These computer codes can collect appropriate information about the characteristics of engineering problems before actually running the process. Computer simulations can provide a rapid investigation of various alternative designs to decrease the required time to improve the system. In addition, most numerical analyses for engineering problems make a well-suited use of mathematical programming. The main goals of simulation include what-if study of a model or sensitivity analysis and optimization and validation of the model [1]. The essential benefit of simulation is its ability to cover complex processes, either deterministic or random, while eliminating mathematical sophistication [2]. Clearly, because of the complexity of mathematical formulation analyzing in many real-world optimization problems, simulation-optimization methods become necessary to find more interest and popularity than other optimization methods [3]–[5].

CPS combine physical objects or systems with integrated computational facilities and data storage [6]. CPS is a key enabling technology in systems intelligence. In CPS embedded computers and networks, the physical processes are controlled usually with feedback loops where physical processes affect computations and vice versa [7]–[10]. CPS are multidimensional and complex systems that integrate the cyber world with the dynamic physical world. Integrating physical processes with computer systems is the main challenge presented in CPS as the computational cyber part continuously senses the state of the physical system and applies decisions and actions for its control [11]. The integration and collaboration of three terms including computing, communication, and control are known as “3C” [12], [13], CPS provides sensing, real-time optimization, information feedback, dynamic control, and other services, see Figure 1. In recent years, the application of CPS has been widely considered in different fields such as aerospace [14]–[16], defense [17], [18], energy systems [19], [20], healthcare [21]–[24], vehicle [25]–[27], and others [28]–[30].

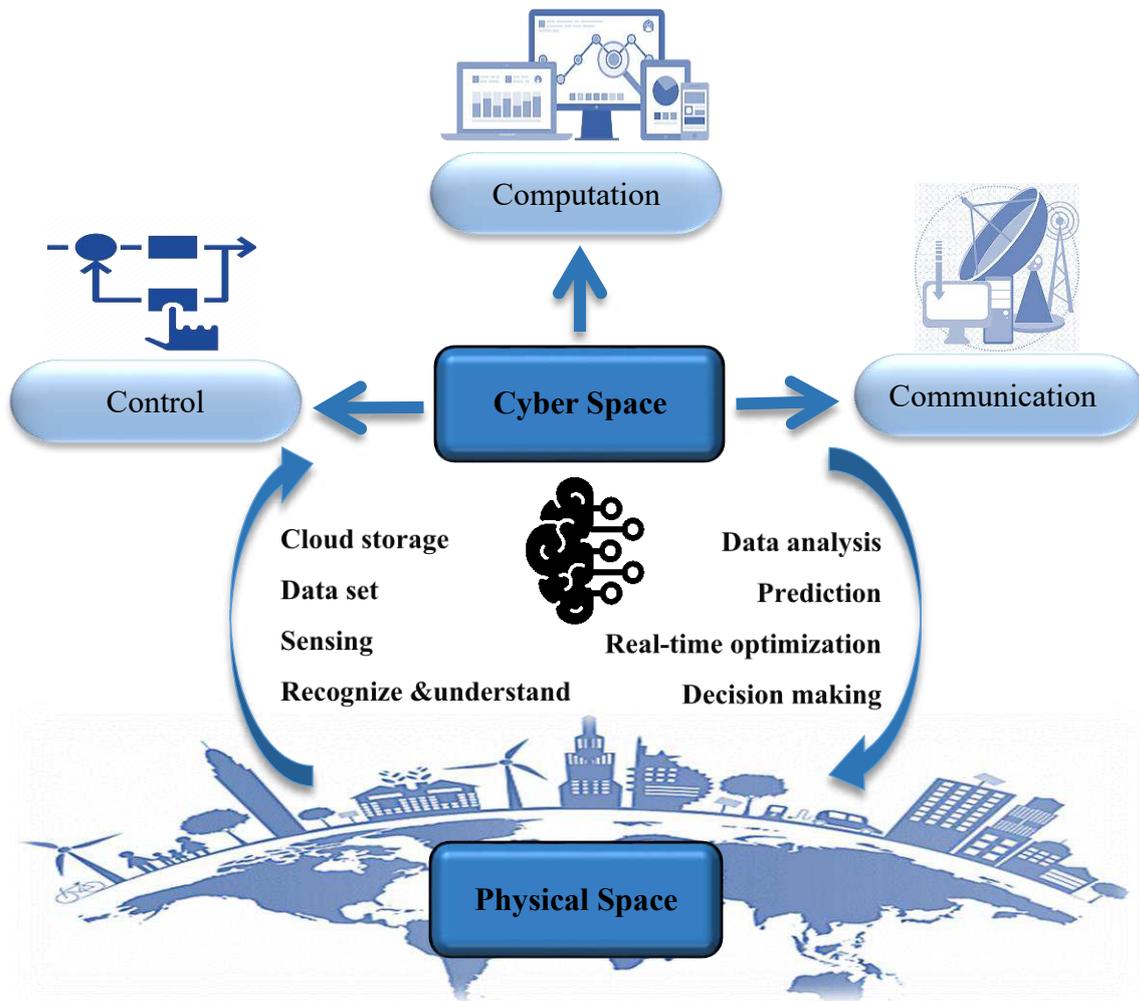


Figure 1 Representation of cyber-physical system (CPS).

In industrial practice, many CPS systems have been designed by decoupling the control system design. In this way, CPS and real-time interaction are achieved in order to monitor and control physical entities in a reliable, safe, collaborative, robust, and efficient way [12], [13]. Using precise calculations to control a seemingly unpredictable physical environment is a great challenge [31]. After the CPS control system is designed and modeled by extensive simulation, tuning methods need to be expanded to address uncertainty and random disturbances in the system. In addition, ignoring the impact of uncertainty on the optimization model, the obtained optimal results may be far from the true optimum settings [32]. One of the main features in a reliable CPS design is stability feature (robustness), which means no matter how the environment generates noise and uncertain factors, the control system should always reach a stable decision

result eventually [33]. Robustness in CPS control system seeks to achieve a certain level of performance with possible modeling errors in the forms of parametric or nonparametric uncertainties [34]. However, considering uncertainty and random disturbances, while keeping the function and operation of the system, has been computationally time-consuming and costly.

Because of uncertainty, more complexity in real-time control implementation of CPS is unavoidable. So, looking for less expensive computational methods of optimization considering uncertainties has become interesting among most engineering applications. To overcome such computational difficulties, researchers have applied surrogate-based learning methods (e.g. polynomial regression, GP, and radial basis function) [35]–[37]. Surrogate-based methods can ‘learn’ the problem behaviors and approximate the function value. These approximation models can accelerate the function evaluation as well as the estimation of the function value with an acceptable accuracy. In addition, they can improve the optimization performance and provide a better final solution. Various types of real-world engineering optimization problems have been developed by applying surrogate-based methods. These optimization problems include dynamic and stochastic control system design, sub-communities in machine learning problems, discrete event systems (e.g. queues, operations, and networks), manufacturing, medicine and biology, engineering, computer science, electronics, transportation, and logistics, see [3], [5], [36], [38]–[40]. However, several studies have systematically illustrated the applications of surrogate-based optimization algorithms [36], [37], [41]–[43].

In this paper, a new outline of robust real-time optimization in the CPS control model under the effect of environmental factors (also known as noise factors or uncertainty, see [4], [5]), and variability in feedback loop due to sensor’s anomaly is studied. The main contributions of this study are as following:

- i. We propose a new hybrid surrogate/metaheuristic method combining the GP surrogate and PSO algorithm. We apply the PSO metaheuristic to update the GP surrogate for sequential investigation of a robust optimal result. The proposed

hybrid GP/PSO algorithm has the advantages of both GP surrogates in learning the behavior of the model in an efficient global optimization with PSO metaheuristic in convergence searching for optimum results.

- ii. We apply the straightforward jackknife leave-one-out technique to estimate surrogate prediction error applied in efficient global optimization. This method can estimate surrogate prediction error apart from the type of surrogate using a training set of sample points.
- iii. An augmented bootstrapping technique is used to analyze the sensitivity of robust optimization results. This technique can apply the initial set of Input/Output (I/O) data instead of resampling the model.
- iv. An exhaustive search method for optimization under uncertainty with a straightforward procedure is developed. It can be easily applied in real operation of the CPS framework. In this study, the proposed algorithm is applied to provide robust tuning of the FOPID controller over two sources of variability. The first source is related to real-time setpoint that is predicted by learning from collected data (e.g. surrogate) over CPS environmental factors and the second variability is found in output's feedback due to anomaly in sensors.
- v. Energy consumption and response error are optimized as a robust multi-objective optimization model by Pareto frontier estimation in the real-time computational part of the CPS model (see Figure 1).

The rest of this paper is organized as follows. Section 2 provides more details about real-time FOPID control when two types of uncertainties (noises) including environmental factors and sensor anomaly are considered in a CPS framework. Materials and methods of the proposed algorithm to handle robust multi-objective optimization of a CPS control system are elaborated in Section 3. In Section 4, the applicability and effectiveness of the proposed approach are examined to provide robustness and reliability in the robust optimal design of the FOPID controller in the CPS framework of a five-bar linkage robot manipulator. Finally, this paper is concluded in Section 5.

2. Problem Statement

The existing uncertainties and anomalies in the cyber environment have resulted in emerging concerns about the traditional control system [34]. In real-time control of CPS, physical process variables are monitored and processed by intelligent controllers for keeping the values of safety parameters between the given thresholds. Environmental conditions can affect the system dynamics and also the controller function [9]. The precision of computing must interface with the uncertainty and the noise in the physical environment [44]. The physical world, however, is not entirely predictable. It is normal that the CPS do not operate in a controlled environment. So, it must be robust to uncertainty (unexpected conditions) and adaptable to subsystem failures [8].

2.1 FOPID controller

In this paper, for better control, fractional-order $PI^\lambda D^\mu$ controller is used. Currently, fractional-order controllers are being extensively used by many scientists in order to achieve the most robust performance of the systems [45]. The main reason for choosing FOPID controllers is their additional degrees of freedom that result in a better control performance [46], [47]. A generalized FOPID controller was first introduced by [48] which proposed $PI^\lambda D^\mu$ controller—involving a λ order integer and a μ order differentiator. The differential equation of a fractional-order $PI^\lambda D^\mu$ controller is defined by:

$$u(t) = K_p e(t) + K_i D_t^{-\lambda} e(t) + K_d D_t^\mu e(t) \quad (1)$$

The reliability of FOPID controller depends on the optimal design of three gain parameters (K_i, K_p, K_d) and two order parameters (λ, μ). However, we try to further increase the reliability of the tuning result by setting the traits of the simulation model to be as close as possible to the practical condition to make a CPS outline for the FOPID controller. The FOPID control system with a single setpoint does not express the aspects of the behavior that are essential to the system in the context of CPS. Moreover, we

challenge the robust control to achieve CPS stability when the uncertainty in environmental conditions is the source of variability of the setpoints in the control system. In addition, an uncertain anomaly in sensors causes the noise (variability) in the control feedback loop. Moreover, we aim to tune the FOPID controller robustly in such a CPS control system with real-time setpoints and noise in the model's feedback. Figure 2 shows the control outline of CPS with real-time setpoints and noise in the model's feedback. The application of the integer-order and fractional-order of the PID controller in CPSs has been studied in [47], [49]–[51].

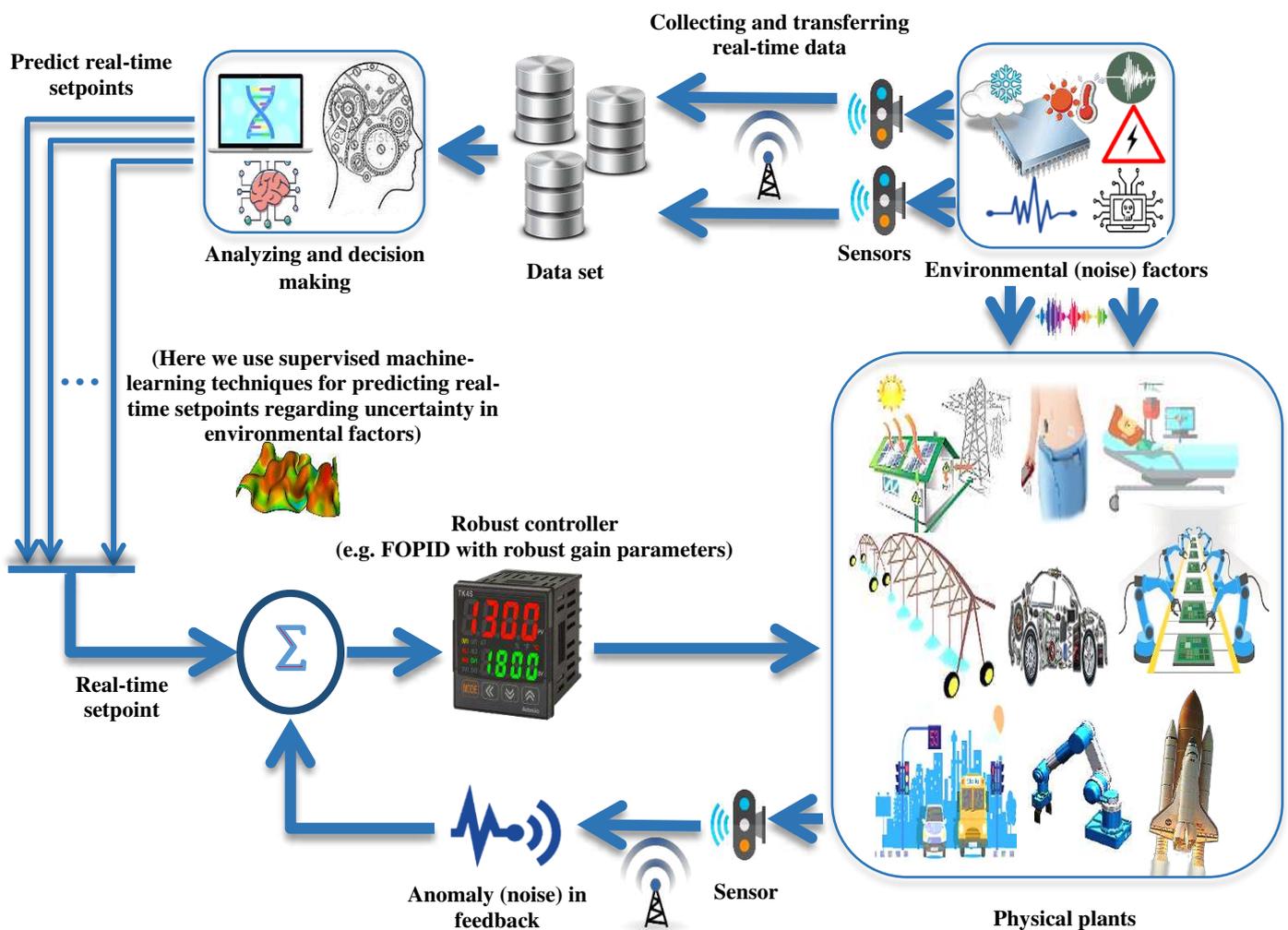


Figure 2 The control framework of CPS with real-time setpoints and noise in model's feedback. The environmental factors would be predicted and applied as a real-time setpoint and anomaly in sensor is estimated in feedback loop. Gain parameters and order parameters in FOPID controller are tuned to be robust against source of variability.

2.2 Uncertainty in the CPS control model

Assume $\tilde{z}_1(t), \tilde{z}_2(t), \dots, \tilde{z}_n(t)$ are the environmental (uncertain) factors in such a CPS control outline. It should be noted that a real-time setpoint of control system at step-time t is affected by variability on the environmental (uncertain) factors. Furthermore, the decision policy needs to be able to predict real-time setpoint in regard to the data collected from the uncertain environmental factors so far. Here, we use supervised learning of data collected so far from environment (e.g. polynomial regression function, $\hat{f}[\tilde{z}_1(t), \tilde{z}_2(t), \dots, \tilde{z}_n(t)]$) and predict the real-time setpoint $\hat{s}(t) = \frac{df}{dt}$, $\hat{s}(t) \in [L_s, U_s]$ in the control system. In addition, an anomaly in sensor to convey response feedback is assumed as an uncertainty that causes the variability in the tuning of the FOPID controller. Assume that the true response of model $y(t)$ is varied by $\tilde{\alpha}\%$ where $\tilde{\alpha}$ is uncertain variable ($\tilde{\alpha} \in [L_\alpha, U_\alpha]$), thus $\tilde{y}(t) = y(t) \times (1 + \tilde{\alpha})$ is a true response that is transmitted to the controller at step-time t . Figure 3 shows a block diagram representation of the CPS control system by considering both types of uncertainty including environmental factors, and sensor anomaly.

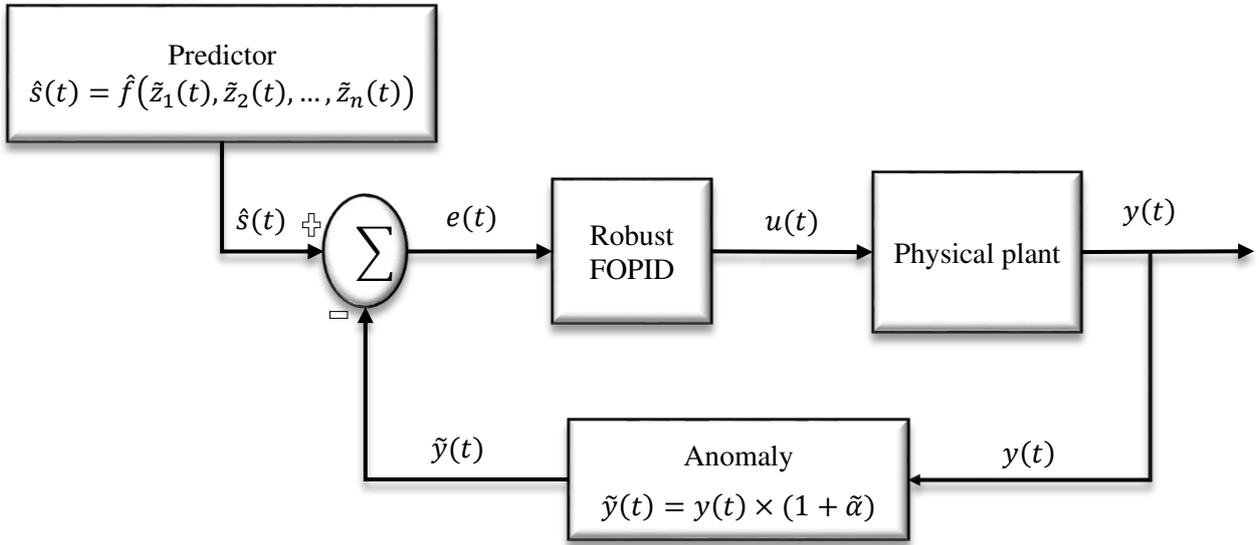


Figure 3 The block diagram of robust FOPID control in CPS framework with real-time setpoints and noise in model's feedback. Real-time setpoint is estimated by approximation function of environmental factors ($\tilde{z}_1(t), \tilde{z}_2(t), \dots, \tilde{z}_n(t)$). Anomaly in sensor's feedback is function of uncertain variable $\tilde{\alpha}$. FOPID gain parameters and order parameters are tuned robustly in such a way to make CPS insensitive against sources of variability in system.

2.3 Objective functions

This study aims at optimizing a robust multi-objective model of the FOPID tuning in the CPS framework by considering two different objective functions (e.g. performance criteria). The first objective function is targeted to manage the Signal Energy Control (SEC) that is consumed in the time domain $0 \leq t \leq T$ as follows:

$$F_1 = \frac{\log(SEC + 1)}{M_1} \quad (2)$$

and

$$SEC = \int_0^T |u(t)|dt = \int_0^T |K_p e(t) + K_i D_t^{-\lambda} e(t) + K_d D_t^\mu e(t)|dt \quad (3)$$

where M_1 is a user-defined big value that is used for normalizing the first objective function in $[0,1]$, so that $M_1 > \log\left(\max_{0 \leq t \leq T} SEC\right)$.

We define the second objective function based on the Response Error Control (REC) (i.e. inspired integrated absolute error) as below:

$$F_2 = \frac{\log(REC + 1)}{M_2} \quad (4)$$

and

$$REC = \int_0^T |e(t)|dt = \int_0^T |\tilde{y}(t) - \hat{s}(t)|dt \quad (5)$$

where M_2 shows a big value that is defined by decision-maker to normalize the second objective function in $[0,1]$, so that $M_2 > \log\left(\max_{0 \leq t \leq T} REC\right)$. Notably, we use a logarithmic scale for both objective functions to smooth the large differences between the values (i.e. cases in which one or a few points are much larger than the bulk of the data). As mentioned earlier, the real-time setpoint $\hat{s}(t)$ in Eq.(5) can be predicted on-time by an easy-to-apply supervised learning like polynomial regression as a function of environmental uncertain factor(s).

2.4 Overall objective function

In order to combine both objective functions including the signal energy control (see Eq.(2)) with response error control (see Eq.(4)) to be used as a single objective model, we apply Lp -mertic approach by $p = 2$ (i.e. for more information about Lp -mertic approach in multi-objective optimization, refer to [52]). Assume $s = (1, 2, \dots, l)$ is the vector of input combination, then we define the Overall Function (OF) as below:

$$OF = \{\theta(F_1)^2 + (1 - \theta)(F_2)^2\}^{\frac{1}{2}}, \text{ for } (s = 1, 2, \dots, l) \quad (6)$$

where θ is a user-defined weighting factor ($0 \leq \theta \leq 1$) that indicates the tendency of the model toward optimization based on each objective function F_1 and F_2 , see Eq.(2) and Eq.(4). Fluctuating this magnitude (θ) provides the capture of Pareto frontier (also called Pareto optimal efficiency) to make a trade-off between each objective function. This approach is a classical method to solve optimization problems when the model is faced with multiple criteria [53]. In fact, the set of optimal solutions obtained from fluctuating θ in $[0, 1]$ provides an estimate of the Pareto frontier.

3. Proposed Algorithm

In this section, we propose a promising technique for optimization under uncertainty using augmented efficient global optimization using the jackknife leave-one-out technique to estimate GP prediction error hybrid GP/PSO method. For this purpose, we first explain the main materials and methods used in the proposed algorithm briefly and then sketch the algorithmic steps in the proposed approach.

3.1 Materials and Methods

3.1.1 Gaussian process (GP) surrogate

GP, that is also known as kriging, is a non-parametric Bayesian approach to supervised learning [54]. GP is an interpolation method that can cover deterministic data and is

highly flexible due to its ability to employ various ranges of correlation functions [55]. In a GP model, a combination of a polynomial model and the realization of a stationary point are assumed by the form of:

$$y = f(X) + Z(X) + \varepsilon \quad (7)$$

$$f(X) = \sum_{p=0}^k \hat{\beta}_p f_p(X) \quad (8)$$

where the polynomial terms of $f_p(X)$ are typically the first or the second-order response surface approach and coefficients $\hat{\beta}_p$ are regression parameters ($p = 0, 1, \dots, k$). This type of GP approximation is called the universal GP, while in the ordinary GP, instead of $f(X)$, the constant mean $\mu = E(y(x))$ is used. The term ε describes the approximation error and the term $Z(X)$ represents the realization of a stochastic process which in general is a normally distributed Gaussian random process with zero mean, variance σ^2 , and non-zero covariance. The correlation function of $Z(X)$ is defined by:

$$Cov[Z(x_k), Z(x_j)] = \sigma^2 R(x_k, x_j) \quad (9)$$

where σ^2 is the process variance and $R(x_k, x_j)$ is the correlation function that can be chosen from different correlation functions which were proposed in the literature (e.g. exponential, Gaussian, linear, spherical, cubic, and spline), see [56], [57]. Today, GP surrogate has been used as a widespread global approximation technique that is applied widely in control engineering design problems [38], [58].

3.1.2 Particle swarm optimizer (PSO)

The canonical PSO algorithm was proposed by [59] and is inspired by the social behavior of swarms such as bird flocking or fish schooling. The parameters of PSO consist of the number of particles, position of agent in the solution space, velocity, and neighborhood of particles (communication of topology). The PSO algorithm begins with initializing the population. The second step is to calculate the fitness values of each particle, followed by updating individual and global bests as the third step. Then,

velocity and the position of the particles become updated (step four). The second to fourth steps are repeated until the termination condition is satisfied [60], [61]. The PSO algorithm is formulated as follows [59]–[61]:

$$v_{id}^{t+1} = v_{id}^t + c_1 \cdot rand(0,1) \cdot (p_{id}^t - x_{id}^t) + c_2 \cdot rand(0,1) \cdot (p_{gd}^t - x_{id}^t) \quad (10)$$

$$\text{and } x_{id}^{t+1} = x_{id}^t + v_{id}^t$$

where v_{id}^t and x_{id}^t are particle velocity and particle position respectively. d is the dimension in the search space, i is the particle index, and t is the iteration number. Expressions c_1 and c_2 represent the speeds of regulating the length when flying towards the most optimal particles of the whole swarm and the most optimal individual particle. The term p_i is the best position achieved by particle i so far and p_g is the best position found by the neighbors of particle i . The expression $rand(0,1)$ shows the random values between 0 and 1. The exploration happens if either or both of the differences between the particle's best (p_{id}^t) and previous particle's position (x_{id}^t), and between population's all-time best (p_{gd}^t) and previous particle's position (x_{id}^t) are large. In addition, the exploitation occurs when these two values are both small. PSO has attracted wide attention in control engineering design problems due to its algorithmic simplicity and powerful search performance [62], [63]. However, PSO algorithm that requires a large number of fitness evaluations before locating the global optimum is often prevented from being applied to computationally expensive real-world problems [64]. Therefore, surrogate-assisted PSO metaheuristic optimization algorithms have been focused in the literature, see [64]–[66].

3.1.3 Uncertainty management

Here, we follow [37], [67], [68] and inspire Taguchi's overview of robust design [69] for dealing with uncertainty as a source of variability in the model. However, we expand Taguchi robust design terminology and apply its definition for environmental noise factors in such a CPS control system. But in this study, we replace statistical approach of Taguchi viewpoint with augmented efficient global optimization using jackknife

leave-one-out technique and hybrid GP/POS approach. Furthermore, we first intersect two sampling design sets. One sampling design is for decision variables (inner array) and another is for uncertain variables (outer array). Given that $s = (1, 2, \dots, l)$ is the vector of sample points over decision variables, and $r = (1, 2, \dots, m)$ is the vector of uncertainty scenarios, so $l \times m$ input combinations are designed, and the real model (or true simulation model) are evaluated $l \times m$ times to collect relevant simulation outputs, see Figure 4. Assume Y is the $l \times m$ matrix of simulation outputs (i.e. in this study the simulation outputs include OF values that can be obtained regarding Eq.(6)), thus mean and standard deviation (Std) for each arrow in Y can be computed by the following equations:

$$Mean_s = \frac{1}{m} \sum_{r=1}^m y_{sr} \quad , for \quad (s = 1, 2, \dots, l) \quad (11)$$

$$Std_s = \sqrt{\frac{1}{m} \sum_{r=1}^m y_{sr}^2 - \left(\frac{1}{m} \sum_{r=1}^m y_{sr} \right)^2} \quad , for \quad (s = 1, 2, \dots, l) \quad (12)$$

Signal-to-Noise Ratio (SNR) as introduced by Taguchi [69], [70] is a robustness criterion based on the mean and the Std of a system response Y . Given that, Y is the smaller the better type, Taguchi assumed zero as the minimal possible response value. Accordingly, he formulated the following SNR as the robustness criterion:

$$SNR_s = 10 \log[Mean_s^2 + \omega * Std_s^2] \quad , for \quad (s = 1, 2, \dots, l) \quad (13)$$

Since we performed a minimization of the model's output (here is overall function, see Eq.(6)) to find the optimal parameters of FOPID controller, the formulation of the SNR in Eq.(13) has the opposite sign by Taguchi formulation. Additionally, a weighting parameter ω is introduced to allow for individual emphasis on the minimization of variations. The smallest value of SNR in Eq.(13) depicts the better point with smaller relevant simulation output and higher insensitivity to the source of variability (robustness).

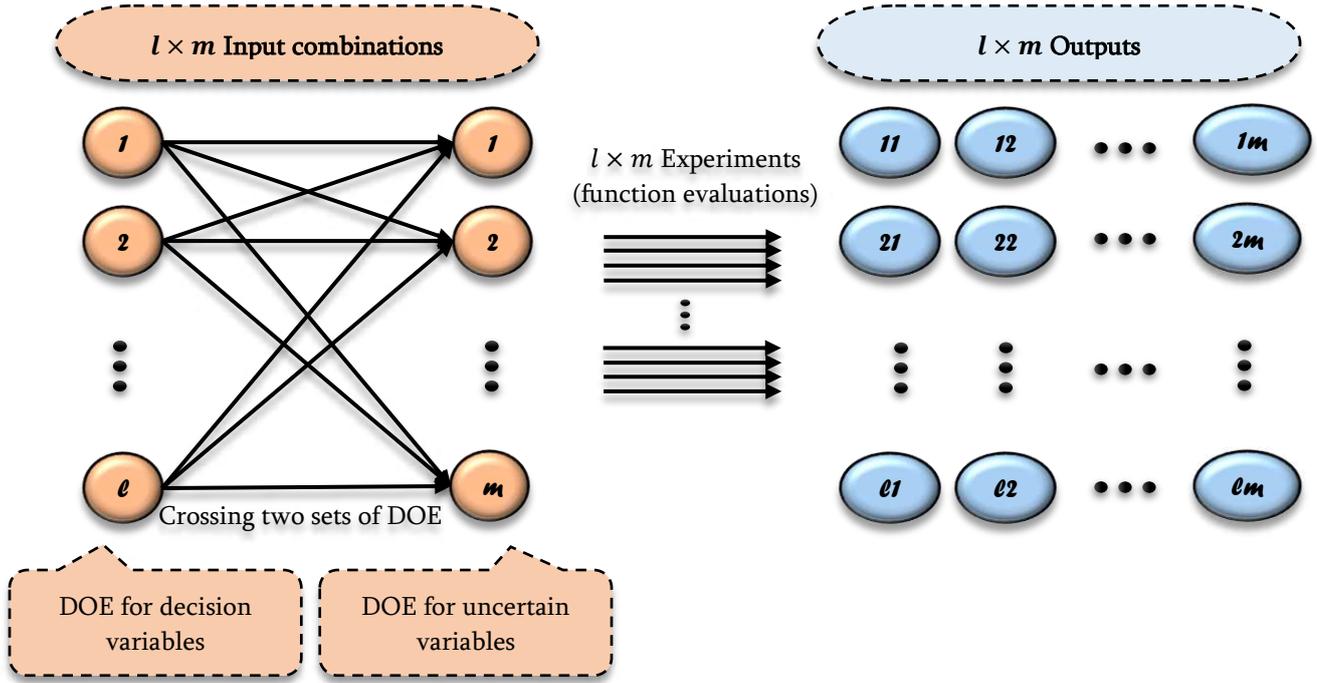


Figure 4 Crossing two sets of DOE dealing with uncertainty in a model, one DOE (l samples) over decision variables of the model and second DOE (m samples) over uncertain variables in the model.

3.1.4 Efficient global optimization using a jackknife leave-one-out strategy

A common formulation of efficient global optimization has been developed in the outline of the expected improvement criterion, see [71], [72]. The expected improvement (EI) method has been developed in engineering design problems to adaptively improve the local and global search of optimal points (i.e. control a trade-off between exploration and exploitation properties) [73]. This method has been combined with two main parts. The first statistical part consists of the design of experiments and surrogate techniques and the second part involves evolutionary algorithms. If SNR_c is considered for the arbitrary point c , an improvement function over the best point that is so far computed with SNR_b is defined as $\max\{0, (SNR_b - SNR_c)\}$. A common formulation of efficient global optimization in term of expected improvement creation is constructed as below:

$$EI(c) = (SNR_b - \widehat{SNR}_c) \Phi\left(\frac{SNR_b - \widehat{SNR}_c}{\hat{S}_c}\right) + \hat{S}_c \phi\left(\frac{SNR_b - \widehat{SNR}_c}{\hat{S}_c}\right) \quad (14)$$

where \hat{S}_c indicates the estimation of GP prediction's error on candidate point c . The expression SNR_b shows the value of the best signal to noise ratio that is obtained from true data of the original simulation model, and \widehat{SNR}_c is GP surrogate prediction on candidate point c . The terms Φ and ϕ depict the cumulative distribution function (CDF) and probability density function (PDF) of a standard normal distribution respectively. The first phrase (Φ) in Eq.(14) is related to the local search and the second phrase (ϕ) is related to a global search. In the search for the next best point among all the candidate points, the point with maximum EI term in Eq.(14) is selected and replaced with the best point so far obtained. This procedure is continued until $Max REI - 0 \leq \varepsilon$, where ε is a user-defined threshold, or an allocated computational cost (e.g. fixed number of repetitions) is reached. However, to find the neighbor points (candidate points) around the current best point, different strategies of sampling design methods can be used such as factorial designs [74] and space-filling design [75]. Here, we apply PSO global optimizer to investigate the maximum EI among the whole design space.

In order to estimate the surrogate prediction error for c th candidate point (\hat{S}_c) in Eq.(14), simulation experiments can be resampled [71], [72]. The authors in [75] have used the bootstrapping technique to obtain prediction error for a GP surrogate using resampling to refit surrogate and obtain prediction error. However, resampling imposes extra computational cost due to the additional number of required simulation experiments (function evaluations). Here, in order to estimate the surrogate prediction error for each candidate point, we apply the jackknife leave-one-out approach. This approach applied an available set of I/O data and doesn't need resampling and extra simulation experiments.

3.1.5 Jackknife leave-one out approach

Jackknife was first introduced by Quenouille (1949) [76] and named by Tukey (1958) [77]. The application of the jackknife method involves a leave-one-out strategy for the estimation of a parameter (e.g. the variance) in a dataset [78]. In this study, we are motivated to use the jackknife leave-one-out approach to estimate surrogate prediction error (\hat{S}_c) required in Eq.(14) formulation, because

this method uses an existing data and does not require to re-run the expensive simulation model. Here, this method is used to predict GP prediction error while it can be used for other surrogates as well. Let \widehat{SNR}_c denotes the prediction of GP surrogate that fitted over all l samples (input combinations), therefore the GP prediction error in c th candidate point (\hat{S}_c) can be estimated through the jackknife leave-one-out approach as the steps in Algorithm 1.

Algorithm 1 Jackknife leave-one-out approach.

Input: Set of input combinations and relevant output (SNR).

Output: Estimation of surrogate prediction error for c th candidate point.

begin

Step 1. Select l_c samples from the complete set of l combinations ($s = 1, 2, \dots, l$) when $l_c = l - k$ and k is a set of samples located in vertices (i.e. we aim to avoid extrapolating of GP surrogate).

Step2: Drop u th samples (simulation experiment) and relevant SNR output when ($u = 1, 2, \dots, l_c$).

Step 3: Fit a new GP surrogate over $(l_c - 1 + k)$ remaining samples.

Step 4: Predict output for c th candidate point (\widehat{SNR}_c^{-u}) using the GP surrogate constructed from the previous step.

Step 5: Implement three previous steps for all l_c samples computing l_c relevant predictions.

Step 6: Apply the jackknife estimator to obtain the estimation of surrogate prediction error for c th candidate point as below:

$$\hat{S}_c \approx \left\{ \frac{1}{l_c} \sum_{u=1}^{l_c} (\widehat{SNR}_c - \widehat{SNR}_c^{-u})^2 \right\}^{1/2}$$

end

3.2 Algorithmic Framework

In this study, we develop a new hybrid surrogate/metaheuristic method applied in robust efficient global optimization and optimization under uncertainty. We apply a PSO metaheuristic to update a GP surrogate for sequential investigation of a robust optimal point. The proposed algorithm can handle robust efficient global optimization by the exhaustive search method that can be applied in real operation of CPS control frameworks. The algorithmic representation of the proposed approach is presented in Figure 5. The main steps involved in the proposed algorithm are presented in Algorithm 2.

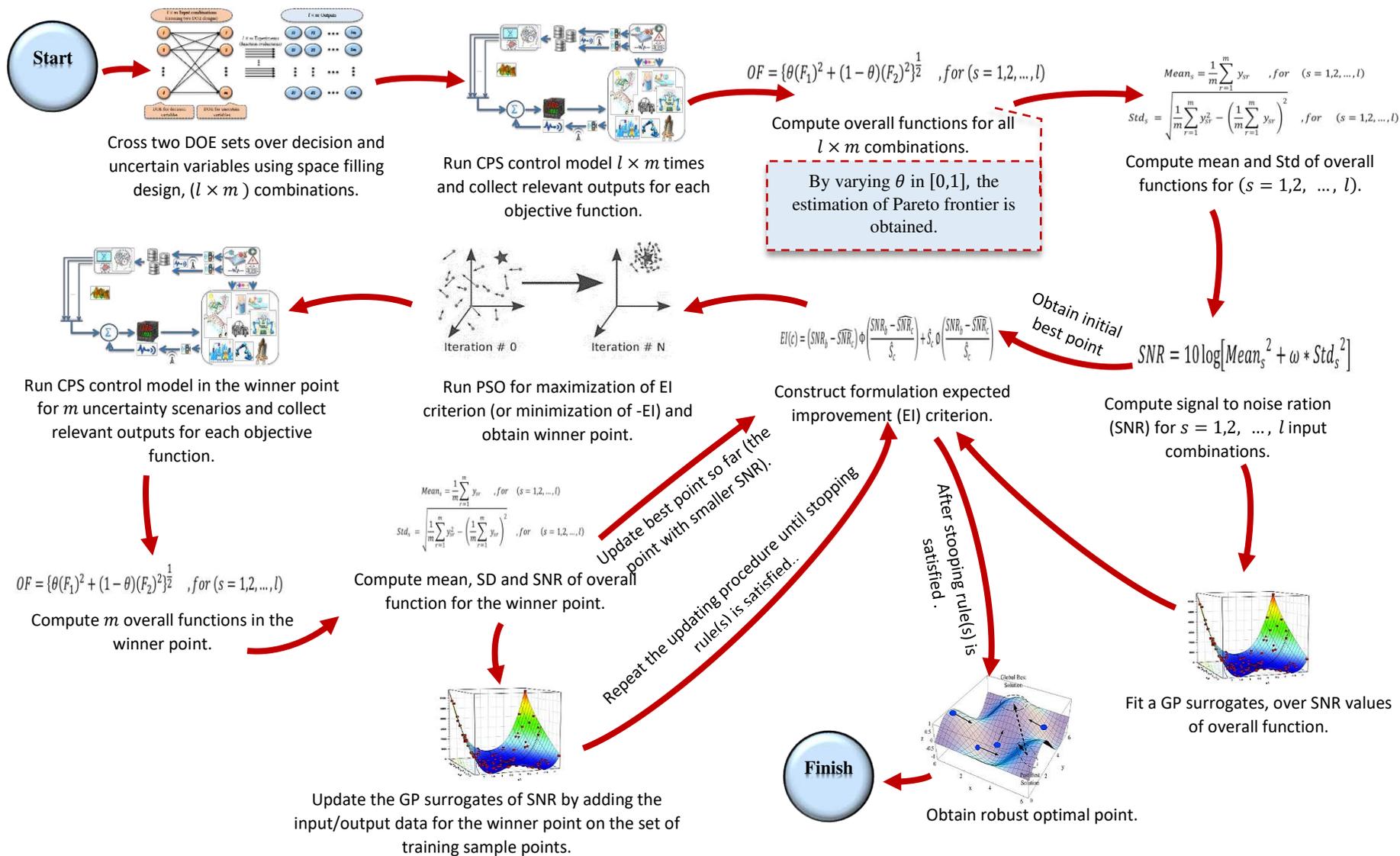


Figure 5 Algorithmic representation of proposed approach for hybrid GP-PSO based robust simulation-optimization under uncertainty.

Algorithm 2 Proposed robust simulation-optimization approach.

Input: Estimated upper (U_s) and lower bound (L_s) for system's setpoint $\hat{s}(t) \in [L_s, U_s]$ and upper (U_α) and lower bound (L_α) for $\tilde{\alpha}$ due to anomaly in sensor feedback.

Output: The estimation of the Pareto frontier by a set of robust optimal points found by the proposed approach.

begin

Step 1. Design crossed array (using the space-filling design) by crossing two sets of experiments with dimensions $l \times m$ as below:

- An inner array matrix with dimension $(1 \times n_x)$ where l is the number of sample points for decision variables and n_x is number of decision variables (e.g. in FOPID tuning $n_x=5$ decision variables including three gain K_i, K_p, K_d and two order λ, μ parameters).
- An outer array matrix with dimension $(m \times n_z)$ where m is the number of sample points (uncertainty scenarios) for n_z uncertain variables (e.g. here in represented CPS control system $n_z = 2$ including $\hat{s}(t)$ and $\tilde{\alpha}$).

Step 2. Run the CPS model (i.e. here we use simulation model) for each crossed $(l \times m)$ combination and obtain the relevant output \hat{y}_{sr} regarding each objective function, when $s = (1, 2, \dots, l)$ and $r = (1, 2, \dots, m)$.

Step 3. Compute overall function (OF) values for all $l \times m$ input combinations using Eq.(6).

Step 4. Compute $Mean_s$ and Std_s of overall function using Eq.(11) and Eq.(12) for each $s = (1, 2, \dots, l)$ sample point in inner array and compute relevant SNR_s using Eq.(13).

Step 5. Fit a GP surrogate over sets of I/O data (with l input combinations and relevant SNR_s values).

Step 6. Define an initial best point among the set of I/O data obtained from Step 4 (the point with the smallest SNR regarding Eq.(13)).

Step 7. Set expected improvement criterion (see Eq.(14)) as an objective function in PSO optimizer algorithm (i.e. with minimizing of $-EI(c)$) and obtaining a winner point.

Step 8. Run the real CPS model (e.g. original simulation model) in the winner point for m combinations of uncertainty (scenarios) designed in Step 1 and obtain relevant outputs for each objective function.

Step 9. Obtain OF values for the winner point regarding m uncertainty scenarios.

Step 10. Compute mean and Std of the winner point using Eq.(11) and Eq.(12).

Step 11. Update the set of I/O data $s = (1, 2, \dots, l + i)$, when i is the number of the sequential runs.

Step 12. Fit a new GP surrogate over an updated set of I/O data (with $l + i$ training points and SNR as outputs).

Step 13. Update (if needed) the best point obtained so far to a point with smallest SNR ratio among all the sample points (including initial training points and points which are added so far for updating of surrogate, see Step 11) and repeat Step 7 till Step 12 until stopping rules are satisfied (e.g. stop sequential updating if $Max EI - 0 \leq \varepsilon$, or $i \leq k$, where ε and k are user-defined thresholds).

Step 14. If stopping rule(s) is satisfied, then set the best point obtained so far as a robust optimal point of the model. The best point so far has the smallest SNR value among all sample points including initial samples and updating sample points.

Step 15. Obtain estimation of Pareto frontier by varying the weight scale θ in $[0, 1]$ (see Eq.(6)) and repeating Step 1 to Step 14.

end

Note that, we assume the approximation function (polynomial regression) fitted over environmental factors $\hat{f}(\tilde{z}_1(t), \tilde{z}_2(t), \dots, \tilde{z}_n(t))$ can be used to estimate upper (U_s) and lower bound (L_s) for $\hat{s}(t)$ by varying $\tilde{z}_1(t), \tilde{z}_2(t), \dots, \tilde{z}_n(t)$ in their relevant ranges (upper and lower bounds of each relevant environmental factor) in time-step t . Here, these bounds are predefined and existed as inputs of the program. In this study, we use a common space-filling design method named Latin hypercube sampling (LHS) with the desired correlational function to design simulation experiments. LHS was first introduced by McKay and colleagues [79]. It is a strategy to generate random sample points while guaranteeing that all the portions of the design space are depicted. LHS has been commonly defined for designing computer experiments based on the space-filling concept. In general, for n input variables, m sample points are produced randomly in m intervals or scenarios (with equal probability). Inspired by [80] in the case of non-independent multivariate input variables, the desired correlation matrix can be used to produce distribution-free sample points in LHS. For more information, refer to [37], [81]. In the represented CPS control system in this study, outputs for two separate F_1 and F_1 objective functions need to be obtained regarding response error control and signal energy control, see Eq.(2) and Eq.(4). Notably, in Step4, each $s = (1, 2, \dots, l)$ sample point is repeated m times through m different combinations (scenarios) of uncertain variables, see the framework of uncertainty management in Section 3.1.3. In Step 7, the fitted GP surrogate over SNR constructed in Step 5 is used to approximate relevant SNR of each search point produced by PSO.

3.3 Sensitivity analysis of optimal results using augmented bootstrapping approach

In this study, the main idea behind the proposed algorithm is to perform sensitivity analysis to expand the information obtained from robust efficient global optimization. Estimating a single optimal point using a particular response may be inaccurate because of variability in the surrogate. Thus, we derive a series of possible responses that take into account a degree of uncertainty by providing confidence regions or prediction

intervals. The author in [82] has mentioned two alternative strategies for bootstrapped resampling as follows:

- In each set of bootstrapping, both sets of input (design) combination (X) and noise (uncertain) combination (Z) are resampled randomly.
- The resampling is adapted to noise or uncertain component (Z) only while keeping the deterministic input combination (X) fixed.

Here, to find the bootstrapped set of data, a model is resampled B times ($b = 1, 2, \dots, B$) (sampling with replacement), while B is the number of resampled or bootstrapped sample size. Moreover, B separate surrogates are fitted on B different sets of sample points with the same size (n design points). It is assumed that d^+ is a robust optimal solution which is obtained from the original (non-bootstrapped) surrogate. All output values in point d^+ are estimated using all the B bootstrapped surrogates. The distribution-free bootstrapped Confidence Intervals (CIs) can be computed as below, [56], [83]:

$$P(d^{+*}_{(\lfloor B(\alpha) \rfloor)} \leq d^+ \leq d^{+*}_{(\lfloor B(1-(\alpha/2)) \rfloor)}) = 1 - \alpha \quad (15)$$

The superscript “*” is a common symbol for bootstrapped values [56]. The expression α gives two-sided CIs. Bonferroni’s inequality suggests that Type I error rate for each interval per output is divided by the number of outputs (here is SNR). If the values of bootstrap estimate $SNR(d^+)^*$ are sorted from low to high, then $\lfloor \cdot \rfloor$ and $\lceil \cdot \rceil$ respectively denote floor and ceiling function to achieve the integer part and round upwards.

In this study, as inspired by [68], [84], the particular augmented bootstrapping approach is used for costly simulation running. In such a case, assume the set of sample points is fixed and only old data to fit surrogate with enough replication is available and new simulation replicating is very expensive. This augmented bootstrapping approach does not imply extra computational cost because of resampling and required simulation running to find a bootstrapped set of data. x_s ($s = 1, 2, \dots, l$) denotes the set of sample points and each x_s is repeated m times ($r = 1, 2, \dots, m$). We assume that the original set

of data obtained from the original simulation model is available (size $l \times m$) when m is the number of scenarios for uncertainty and l is the number of input combinations. Moreover, the augmented bootstrapping procedure is sketched in Algorithm 3.

Algorithm 3 The augmented bootstrapping procedure.

Input: Set of I/O data, and robust optimal point.

Output: Estimation of CIs.

begin

Step 1. Set $s = 1$ and $r = 1$.

Step 2. Choose (with replacement) one random number from the collection of $\{r^* = 1, 2, \dots, m\}$.

Step 3. Replace the r th original output $y_{s,r}$ (selected from the old data) with the bootstrap output $y_{s,r}^* = y_{s,r^*}$.

Step 4. Set $r = r + 1$ and continue Step 3 and Step 4 till $r = m$.

Step 5. Set $s = s + 1$ and continue Step 3, Step 4 and Step 5 till $s = l$.

Step 6: Compute $Mean_s^*$, Std_s^* , and SNR^* using Eq.(11), Eq.(12), and Eq.(13) respectively for $(s = 1, 2, \dots, l)$ and fit a GP surrogate over new set of I/O data.

Step 7: Continue resampling B times ($b = 1, 2, \dots, B$) where B is the number of resampling or bootstrap sample size and compute $SNR_b^* = (SNR_1^*, SNR_2^*, \dots, SNR_b^*)$.

Step 8: Compute bootstrapped CIs using Eq.(15) for the robust optimal point obtained by the proposed algorithm as elucidated in Section 3.2).

end

Note that, regarding the Step 1 till Step 5, it can be seen that a random number with replacement in $[1, m]$ is selected and regarding the selected number, we choose the relevant response in an outer array (see the structure of the crossed array design explained in Section 3.1.3) that was previously collected from original simulation model and has the same column number. For the same input combination, we repeat this procedure m times and collect m different responses or may have the same responses (i.e. because the random selection is done with replacement). This procedure is also repeated for other input combinations. Therefore, the data matrix with l row and m column is constructed.

4. Numerical Example

Here, the proposed algorithm is specified for the robust optimal design of FOPID controller in CPS control of five-bar linkage robot manipulators. In the continue, we first explain the dynamics of the five-bar linkage robot manipulators. Next, the robust

optimal design of FOPID controller in the CPS framework of a five-bar linkage robot manipulator is obtained using the proposed algorithm in this paper.

4.1 Dynamics of five-bar linkage robot manipulator

Robotic manipulators, a classic examples of nonlinear systems, are extensively used in the industry to automate various aspects of the production process of goods, thereby improving the quality of human life [85]. With the changing dynamics of these manipulators and their increasing complexity arising from their greater use, there has been a considerable interest in their control technique fields. Robotic manipulators are Multi-Input Multi-Output (MIMO) systems with highly coupled nonlinear dynamics, posing a challenge to the development of their control scheme [86]. A five-bar linkage manipulator is a special class of parallel manipulators where a minimum of two kinematic chains control the motion of end-effectors [87]. The mechanism of a five-bar linkage is shown in Figure 6 [88]. Clearly, even though there are four links being moved, there are in fact only two degrees-of-freedom that are defined as q_1 and q_2 . q_i and τ_i are the joint variable and torque of the i th motor respectively. Likewise, I_i , l_i , l_{ci} , and m_i are the inertia matrix, length, distance to the center of gravity, and mass of the i th link respectively. In addition, if $m_3 l_2 l_{c3} = m_4 l_1 l_{c4}$, then the inertia matrix is

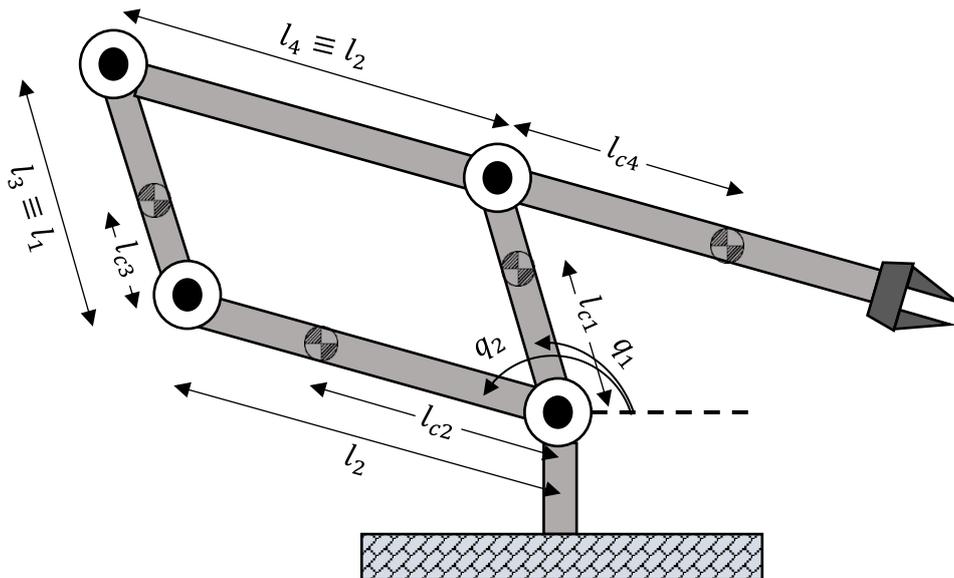


Figure 6 Five bar linkage robot manipulator.

diagonal and constant. As a consequence, the dynamic model of the manipulator is derived by the following equations [88]:

$$\begin{aligned}\tau_1 &= d_{11}\dot{q}_1 + g \cos q_1(m_1l_{c1} + m_3l_{c3} + m_4l_1) \\ \tau_2 &= d_{22}\dot{q}_2 + g \cos q_2(m_1l_{c1} + m_3l_{c3} + m_4l_1)\end{aligned}\tag{16}$$

where g is gravitational constant and d_{11} and d_{22} are as follows:

$$\begin{aligned}d_{11} &= m_1l_{c1}^2 + m_3l_{c3}^2 + m_4l_1^2 + I_1 + I_3 \\ d_{22} &= m_2l_{c2}^2 + m_4l_{c4}^2 + m_3l_2^2 + I_2 + I_4\end{aligned}\tag{17}$$

It should be noted that τ_1 depends only on q_1 but not on q_2 . Similarly, τ_2 depends only on q_2 but not on q_1 . This discussion helps to explain the popularity of the parallelogram configuration in industrial robots. If $m_3l_2l_{c3} = m_4l_1l_{c4}$, then two angles q_1 and q_2 can be adjusted independently without worrying about interactions between the two angles.

4.2 Robust FOPID tuning in CPS control of five-bar linkage robot manipulator

Here, the main goal is to obtain a robust optimal design of FOPID controller in such a CPS control model as elucidated in Section 2. We simulate the five-bar linkage robot manipulator using Eq.(16) in Matlab®/Simulink environment. Simulink does not have a library for the FOPID. Therefore, the controller from the library of FOMCON: a Matlab® toolbox for fractional-order system identification and control [89], which allows for the computation of the fractional-order derivative and integration is used. Numeric values of the parameters of the five-bar manipulator dynamics are taken from [62], [90] as shown in Table 1. From the data driven in Table 1, it is revealed that $m_3l_2l_{c3} = m_4l_1l_{c4}$, thus we can perform robust optimal design of controller for one motor, where the same results are also valid for the second motor. In FOPID tuning, five decision variables including K_i, K_p, K_d, λ , and μ are considered as decision variables. The search procedure for the robust optimal result is done in the ranges as [62]:

$$K_p \in [0,30], K_i \in [0,5], K_d \in [0,5], \mu \in [0,1] \text{ and } \lambda \in [0,1].$$

Table 1 Numeric values of the parameters of the five-bar manipulator dynamics.

Link	Mass (Kg)	Length (m)	C of g (m)	Inertia (Kgm ²)
1	0.2880	0.33	0.166	1
2	0.0324	0.12	0.060	2
3	0.3702	0.33	0.166	1
4	0.2981	0.45	0.075	2

Two performance criteria are considered as outputs of the model including Eq.(2) and Eq.(4) in time-step t (here the size of time-step is fixed at 0.01) and time domain (simulation time) $T = 20$. In addition, for uncertain variables, we assume that $\hat{s}(t)$ varies in $[0.5, 2.5]$ and $\tilde{\alpha}$ varies in $[-0.05, 0.05]$. However, we implement the proposed algorithm in CPS control framework of a five-bar linkage robot manipulator.

The following process is done to determine the robust optimal values of the FOPID parameters (K_i, K_p, K_d, λ , and μ) using the proposed algorithm. First, we design a set of experiments with the size of $l = 15$ samples using LHS. Another sampling design is constructed for uncertain factors $\hat{s}(t)$ and $\tilde{\alpha}$ (here we choose $m = 9$ samples as a size of uncertainty scenarios). Two Matlab® functions “lhsdesign” and “gridsamp” are used in order to design training sample points with minimum correlation and to design uncertainty scenarios (different combinations of uncertain factors) respectively. We cross both sets of experiments to follow crossed array design framework as elaborated in Section 3.1.3. Each input combination in the inner array $s = (1, 2, \dots, l = 15)$ including designed values of K_i, K_p, K_d, λ , and μ are sent to Simulink block for m times regarding each uncertainty scenarios $r = (1, 2, \dots, m = 9)$ and the values of SEC and REC in time domain are collected. So, 15×9 simulation outputs are collected according to 135 simulation runs (function evaluations). We, use the collected data to obtain F_1 , F_2 , and, OF regarding Eq.(2), Eq.(4), and Eq.(6) respectively. We set both M_1 and M_2 parameters equal to 10 used in Eq.(2) , Eq.(4). Regarding m uncertainty scenarios, we repeat each input combination m times, and compute the relevant mean and Std of OF using Eq.(11) and Eq.(12). Then, we calculate SNR_s for each input combination $s = (1, 2, \dots, l = 15)$ using Eq.(13) while assume $\omega = 3$. Afterwards, we fit a GP surrogate

over set of input combinations and set of SNR_s outputs. The DACE [91], Matlab® toolbox has been employed to construct GP surrogate. In the current study, first-order polynomial regression and Gaussian correlation functions are adjusted to fit GP surrogate. The correlation parameter is fix on 0.1 (i.e. in the DACE toolbox, the correlation parameter is forced to vary in the range between 0.01 to 20).

Next, we perform the procedure of sequential expected improvement to estimate the robust optimal point after n sequential EI iterations. Among all the SNR values in the set of SNR_s , a sample with the smallest SNR value and its relevant input combination including the relevant vector of $[K_i, K_p, K_d, \lambda, \mu]$ is considered as an initial best point. Regarding our proposed algorithm, we apply PSO optimizer to search a winner point in each sequential EI iteration. For setting the PSO parameters, the maximum iteration number is fixed 200 and the swarm is initialized with 30 particles. Notably, as we use GP surrogate instead of the true (original) simulation model as an objective function in PSO, thus we don't worry about the computational cost due to running of a true simulation model. At the end of any sequential EI iteration, the program checks the stopping rule(s). Here, we stop the EI procedure when EI criterion becomes smaller than 0.01, or the number of sequential runs reaches 15 iterations. In addition, at the end of any sequential EI iteration, two terms of the program are updated, i) the set of training sample points by adding a winner point and relevant SNR output that is computed accordingly from the original simulation model, ii) the best sample point obtained so far with the smallest SNR among all the training points and updating points. Moreover, according to an updated set of training samples, a new GP surrogate is constructed after each sequential EI iteration. It is important to note that we avoid extrapolation of GP surrogate in each sequential iteration by setting two different rules, i) we consider a death penalty for any point that is investigated by PSO and is located out of bounds of training points, ii) in order to estimate GP prediction error using jackknife leave-one-out approach, we only remove input combination's rows that don't locate on the margin of design space (see Section 3.1.5). The obtained results on performing the proposed algorithm are discussed in the following section.

4.3 Results and discussion

We perform the proposed algorithm for three different values of $\theta = 0.25$, $\theta = 0.5$, and $\theta = 0.75$ in computing OF (see Eq.(6)). In addition, to evaluate the effect of randomness in sampling design methods, each optimization set was repeated 5 times. Table 2, Table 3, and Table 4 show the obtained results using the proposed algorithm for estimating robust FOPID optimal design over five different repetitions for $\theta = 0.25$, $\theta = 0.5$, and $\theta = 0.75$ respectively. It should be noted that in those tables, two expressions “In.sa” and “Up.sa” indicate the initial sampling design and updating samples that are added to the training set through procedure of sequential improvement respectively. As can be seen, for $\theta = 0.25$, the best SNR (-20.621) is obtained through the fifth repetition with a total of 207 function evaluations (15×9 runs regarding initial crossed sampling design and 8×9 simulation runs regarding sequential updating of the training sample set). For $\theta = 0.5$, the best SNR (-22.249) is obtained from the second repetition and with a total of 261 simulation experiments (135 initial samples plus 126 updating samples). The best SNR value (-24.238) for $\theta = 0.75$ is obtained through the second repetition by a total of 270 function evaluations (15 initial input combinations and 15 update combinations that are crossed by 9 uncertainty scenarios). We consider all the three best points over all five repetitions as robust optimal points for the FOPID controller using the proposed algorithm for each $\theta = 0.25$, $\theta = 0.5$, and $\theta = 0.75$ separately. Figure 7 shows the magnitudes of the EI criterion and the best SNR obtained by sequential expected improvement over five different repetitions of the proposed algorithm for $\theta = 0.25$, $\theta = 0.5$, and $\theta = 0.75$. In addition, mean and Std of OF related to the best point so far (smaller SNR) in each sequential EI iteration are shown in Figure 8. It should be considered that two stopping rules are adjusted. The rules mention that EI value becomes smaller than 0.01, or reaches 15 sequential iterations. Figure 9 includes the step responses of the robot manipulator with 9 different uncertainty scenarios ($\hat{s}(t) = [0.5, 1.5, 2.5]$ and $\tilde{\alpha} = [-0.05, 0, +0.05]$) for $\theta = 0.25$, $\theta = 0.5$ and $\theta = 0.75$

Table 2 Robust FOPID optimal results using proposed algorithm over five different repetitions for $\theta = 0.25$.

Repeat No	FOPID Optimal Parameters					Simulation Experiments			Optimum SNR value		
	K_p	K_i	K_d	μ	λ	Total	In.sa	Up.sa	SNR ($\omega = 3$)	Ave	Std
1	4.133	1.411	1.839	0.967	0.967	171	135	36	-20.564		
2	1.000	2.638	2.925	0.033	0.790	180	135	45	-20.209		
3	3.321	1.911	3.875	0.033	0.804	198	135	63	-19.911	-20.290	0.299
4	1.000	4.494	2.854	0.875	0.831	171	135	36	-20.143		
5	1.000	2.250	2.559	0.387	0.852	207	135	72	-20.621		

Table 3 Robust FOPID optimal results using proposed algorithm over five different repetitions for $\theta = 0.5$.

Repeat No	FOPID Optimal Parameters					Simulation Experiments			Optimum SNR value		
	K_p	K_i	K_d	μ	λ	Total	In.sa	Up.sa	SNR ($\omega = 3$)	Ave	Std
1	3.321	1.549	2.634	0.967	0.967	225	135	90	-21.838		
2	1.000	0.713	2.017	0.802	0.726	261	135	126	-22.249		
3	1.000	2.668	3.604	0.575	0.832	225	135	90	-21.879	-22.007	0.173
4	1.000	1.228	2.436	0.657	0.844	189	135	54	-22.120		
5	1.000	2.667	3.947	0.696	0.757	225	135	90	-21.949		

Table 4 Robust FOPID optimal results using proposed algorithm over five different repetitions for $\theta = 0.75$.

Repeat No	FOPID Optimal Parameters					Simulation Experiments			Optimum SNR value		
	K_p	K_i	K_d	μ	λ	Total	In.sa	Up.sa	SNR ($\omega = 3$)	Ave	Std
1	8.238	2.300	3.748	0.426	0.919	261	135	126	-23.868		
2	1.267	3.101	3.332	0.525	0.835	270	135	135	-24.238		
3	4.285	1.998	3.510	0.612	0.938	189	135	54	-24.032	-24.096	0.149
4	7.803	3.239	3.257	0.944	0.918	216	135	81	-24.191		
5	1.025	2.144	3.072	0.567	0.860	270	135	135	-24.151		

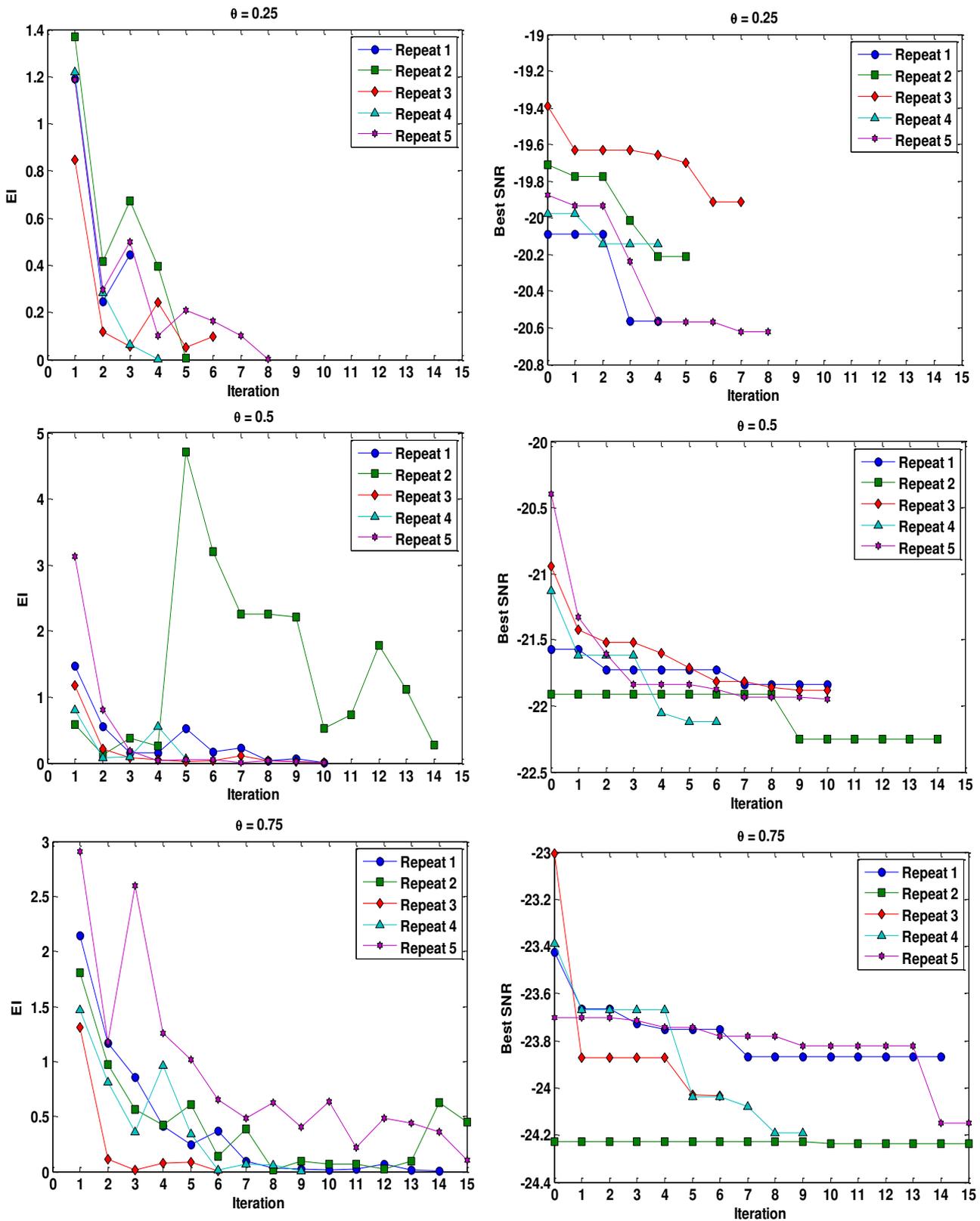


Figure 7 EI criterion magnitudes and best SNR obtained by sequential expected improvement over five different repetition of proposed algorithm for $\theta = 0.25$, $\theta = 0.5$ and $\theta = 0.75$. Two stopping rules are adjusted, EI value becomes smaller than 0.01 or reach 15 sequential iterations.

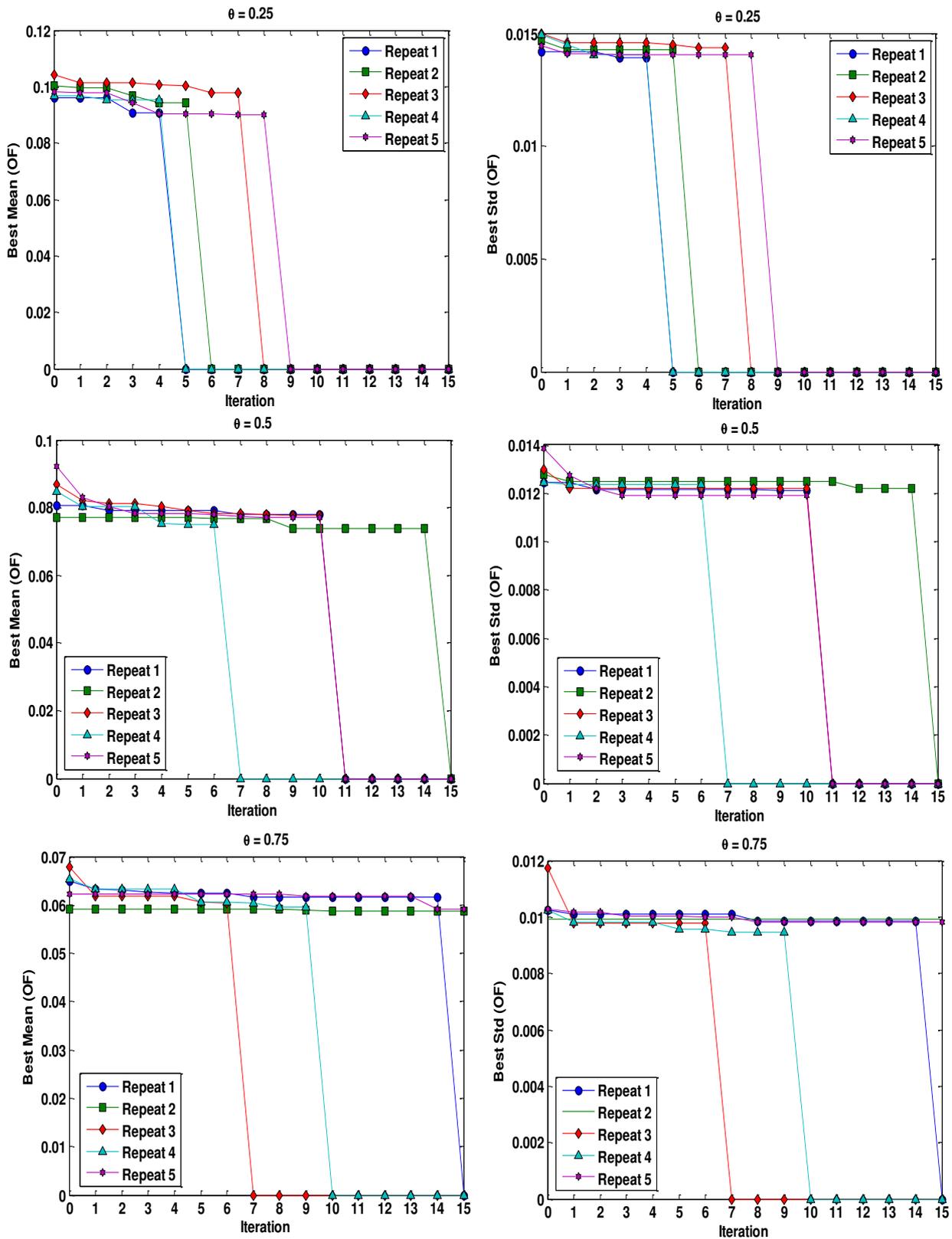


Figure 8 Mean and Std of overall function (OF) related to best point so far (smaller SNR) obtained by sequential expected improvement over five different repetition of proposed algorithm for $\theta = 0.25$, $\theta = 0.5$ and $\theta = 0.75$. Two stopping rules are adjusted, EI value becomes smaller than 0.01 or reach 15 sequential iterations.

Then, we compare the results obtained by a proposed algorithm with three common FOPID optimization methods in the literature ([47], [62], [92], [93]) including the PSO metaheuristic [59] (i.e. when directly used in optimization procedure), Grey Wolf Optimizer (GWO) [94] and Ant Lion Optimizer (ALO) [95]. To make a fair comparison, we set the maximum iteration for all the three mentioned optimizing methods equal to the number of function evaluations (simulation experiments) to obtain a robust optimal point using the proposed algorithm for $\theta = 0.25$, $\theta = 0.5$, and $\theta = 0.75$. To run each of the three mentioned optimizers in each relevant iteration by optimizer, we randomly (with replacement) produce a scenario of uncertainty and compute output of the original simulation including SEC and REC and compute OF as an objective (fitness) function of optimizer. To compare the obtained results using proposed algorithm and the three global optimizers (PSO, GWO, and ALO), we produce 225 different combinations (scenarios) of two uncertain factors including $\hat{s}(t)$ and $\tilde{\alpha}$ using grid sampling design approach. Afterwards, for each set of optimal FOPID parameters according to the obtained results by proposed algorithm and three global optimizers, we run true simulation model regarding each uncertainty scenario (total 225 simulation runs for each set of FOPID optimal point). Table 5, Table 6, and Table 7 include the results of comparison using proposed algorithm and three global optimizers. As can be seen, the proposed algorithm can obtain smaller SNR (more robustness) comparing to others. In addition, the proposed algorithm obtains smaller mean and Std of overall functions for $\theta = 0.25$, $\theta = 0.5$, and $\theta = 0.75$ when it shows the effectiveness of the proposed algorithm in making a tradeoff between both REC and SEC objectives. Figure 9 indicates the variability of the model in each optimal point among simulation outputs including OF, REC, and SEC. Variability in the CPS control model is observed because of uncertainty (source of variability). In Figure 10, the variability of the model is evaluated over 225 different combinations (scenarios) of both uncertain factors including $\hat{s}(t)$ and $\tilde{\alpha}$.

Table 5 Obtained results for FOPID tuning using different methods over 200 different uncertainty scenarios for $\theta = 0.25$.

Method	FOPID Optimal Parameters					SNR ($\omega = 3$)	OF		REC		SEC	
	K_p	K_i	K_d	μ	λ		Mean	Std	Mean	Std	Mean	Std
Proposed algorithm	1.000	2.250	2.559	0.387	0.852	-20.932	0.0882	0.0098	1.2408	0.3652	9.2324	2.7348
PSO	0.000	3.531	1.598	0.000	1.000	-20.811	0.0893	0.0102	1.8751	0.5974	9.1822	2.7483
GWO	2.797	0.992	1.607	0.347	1.000	-20.857	0.0889	0.0101	1.5995	0.4784	9.2232	2.7804
ALO	1.133	3.843	3.210	0.641	0.748	-20.698	0.0907	0.0098	1.0401	0.4044	10.0701	3.0104

Table 6 Obtained results for FOPID tuning using different methods over 200 different uncertainty scenarios for $\theta = 0.5$.

Method	FOPID Optimal Parameters					SNR ($\omega = 3$)	OF		REC		SEC	
	K_p	K_i	K_d	μ	λ		Mean	Std	Mean	Std	Mean	Std
Proposed algorithm	1.000	0.713	2.017	0.802	0.726	-22.663	0.0719	0.0091	1.4563	0.4938	7.7984	2.3618
PSO	0.000	1.282	1.389	0.000	1.000	-20.968	0.0863	0.0135	4.3185	1.6932	8.5815	2.7418
GWO	0.037	0.946	1.904	0.505	0.680	-22.431	0.0736	0.0100	2.0995	0.6635	7.3104	2.2983
ALO	3.163	3.589	3.149	0.421	0.637	-21.202	0.0854	0.0099	1.2341	0.4245	13.7167	4.4300

Table 7 Obtained results for FOPID tuning using different methods over 200 different uncertainty scenarios for $\theta = 0.75$.

Method	FOPID Optimal Parameters					SNR ($\omega = 3$)	OF		REC		SEC	
	K_p	K_i	K_d	μ	λ		Mean	Std	Mean	Std	Mean	Std
Proposed algorithm	1.267	3.101	3.332	0.525	0.835	-24.560	0.0578	0.0072	0.9110	0.2945	10.2119	3.0216
PSO	4.798	0.807	2.493	0.000	1.000	-23.819	0.0630	0.0079	1.2124	0.3929	11.7641	3.5649
GWO	2.539	3.712	2.658	1.000	0.832	-24.042	0.0607	0.0093	1.2848	0.5917	9.9123	2.9452
ALO	0.253	2.279	2.820	0.734	0.795	-24.060	0.0602	0.0100	1.4593	0.7093	8.7589	2.6220

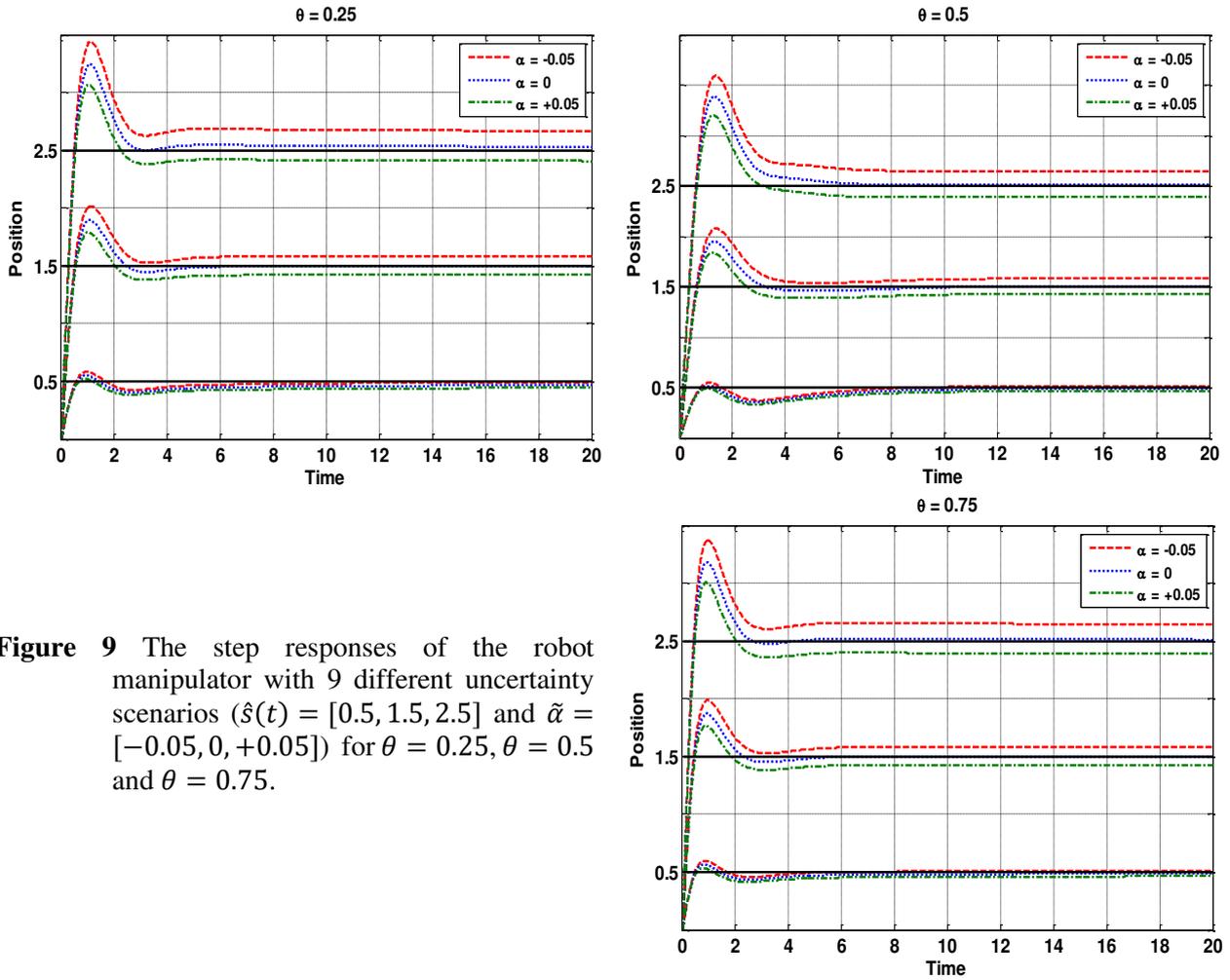


Figure 9 The step responses of the robot manipulator with 9 different uncertainty scenarios ($\hat{s}(t) = [0.5, 1.5, 2.5]$ and $\tilde{\alpha} = [-0.05, 0, +0.05]$) for $\theta = 0.25$, $\theta = 0.5$ and $\theta = 0.75$.

In order to analyze the sensitivity of robust optimal results obtained by the proposed algorithm and estimate the variability which occurred due to randomness in designing sample points, we used the augmented bootstrapping method explained in Section 3.3. Here, based on the obtained results from original GP surrogate, the FOPID parameters in robust optimum point (d^+) for $\theta = 0.25$, $\theta = 0.5$, and $\theta = 0.75$ are defined as below:

- For $\theta = 0.25$

$$d^+ = \{K_p = 1.00, K_i = 2.250, K_d = 2.259, \mu = 0.387 \text{ and } \lambda = 0.852\}$$

$$SNR(d^+) = -20.621$$

- For $\theta = 0.5$

$$d^+ = \{K_p = 1.00, K_i = 0.713, K_d = 2.017, \mu = 0.802 \text{ and } \lambda = 0.726\}$$

$$SNR(d^+) = -22.249$$

- For $\theta = 0.75$

$$d^+ = \{K_p = 1.267, K_i = 3.101, K_d = 3.332, \mu = 0.525 \text{ and } \lambda = 0.835\}$$

$$SNR(d^+) = -24.238$$

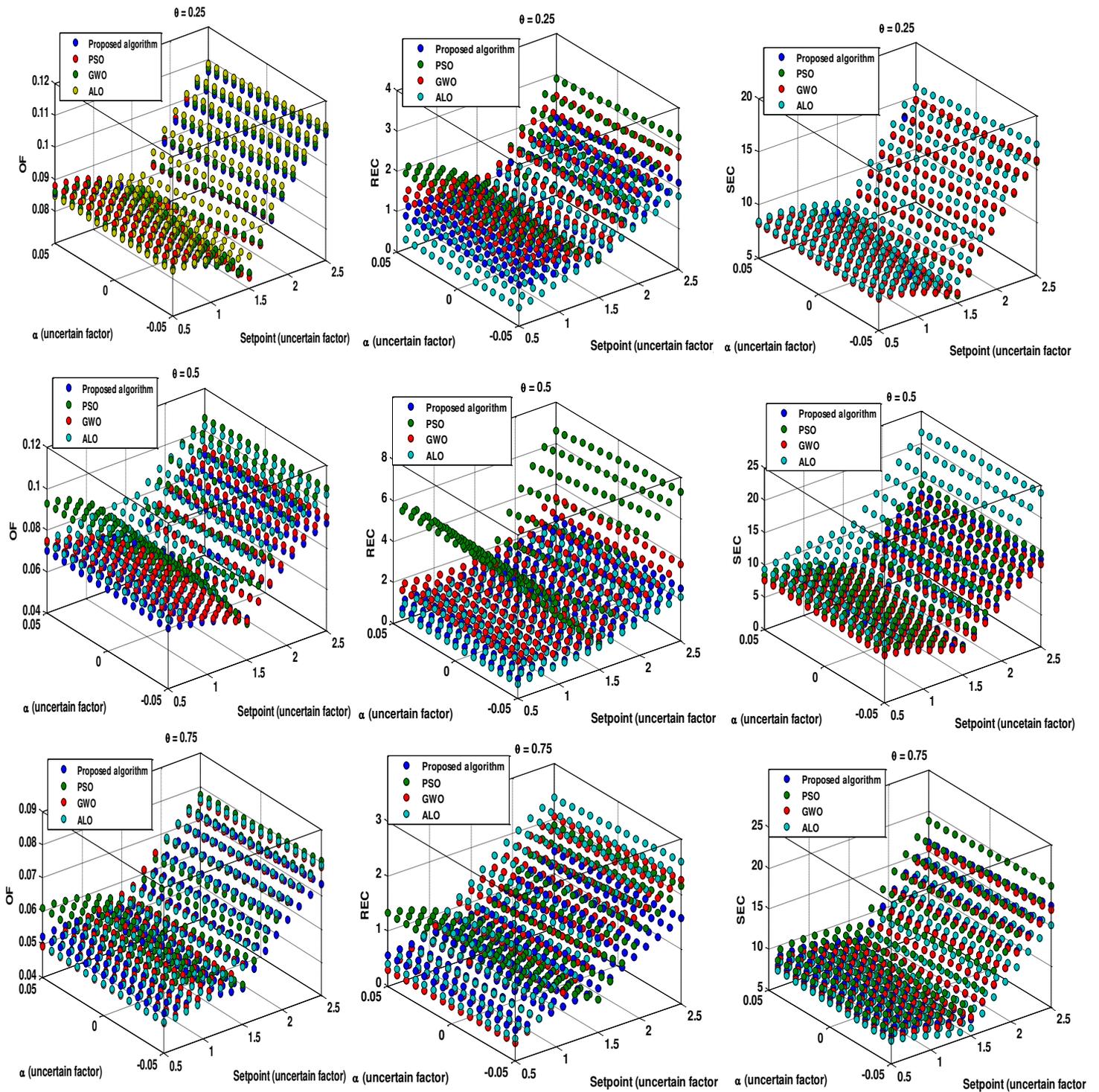


Figure 10 Comparison of proposed algorithm with three common global optimizer in obtained robust optimal point by each method for $\theta = 0.25$, $\theta = 0.5$ and $\theta = 0.75$ dealing with 200 different uncertainty scenarios.

With B predicted values from B bootstrapped GP surrogates, we can quantify the CIs (bootstrapped confidence intervals) for d^+ . For the current case, we selected the bootstrapped size $B = 50$. We predicted SNR by each $B = 50$ bootstrapped surrogates in the robust optimal point which are obtained by original GP surrogates. From these 50 values for SNR, we estimated CIs for SNR by applying Eq.(15). We quantified these confidence regions for $\alpha = 0.05$ (i.e. α denotes type I error and shows the probability of becoming infeasible from estimated confidence regions). As we estimated the robustness as a consequence of the uncertainty in the model, it becomes important to implement further analyses of the statistical variation. The CIs shows that the original estimated SNR may still give variety regarding its threshold due to the variability of surrogates' predictions. However, 95% two-sided approximations of CIs obtained by bootstrapped GP surrogates for SNR values in robust optimal points of FOPID controller regarding $\theta = 0.25$, $\theta = 0.5$, and $\theta = 0.75$ are as follows:

$$P\left(E(d^+)_{(150(0.05/2))}^* \leq E(d^+) \leq E(d^+)_{(150(1-(0.05/2)))}^*\right) = 0.95$$

- For $\theta = 0.25$

$$\text{Lower bound: } E(d^+)_{(150(0.05/2))}^* = -21.566$$

$$\text{Upper bound: } E(d^+)_{(150(1-(0.05/2)))}^* = -20.060$$

- For $\theta = 0.5$

$$\text{Lower bound: } E(d^+)_{(150(0.05/2))}^* = -23.431$$

$$\text{Upper bound: } E(d^+)_{(150(1-(0.05/2)))}^* = -21.585$$

- For $\theta = 0.75$

$$\text{Lower bound: } E(d^+)_{(150(0.05/2))}^* = -25.152$$

$$\text{Upper bound: } E(d^+)_{(150(1-(0.05/2)))}^* = -23.517$$

Figure 11, Figure 12, and Figure 13 show the bootstrapping results for $\theta = 0.25$, $\theta = 0.5$, and $\theta = 0.75$ respectively regarding FOPID parameters including gain parameters (K_p , K_i , K_d) and order parameters (μ , λ).

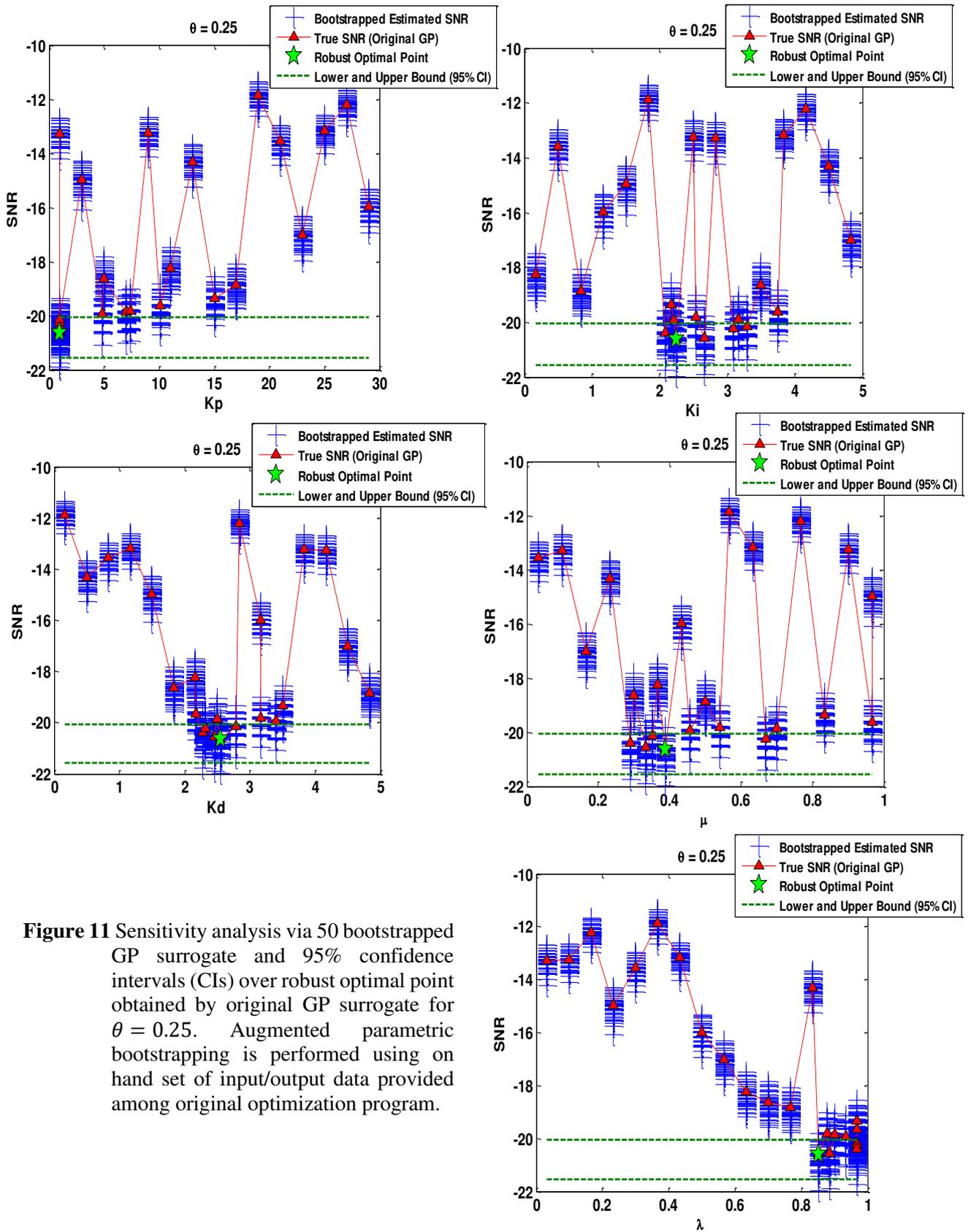


Figure 11 Sensitivity analysis via 50 bootstrapped GP surrogate and 95% confidence intervals (CIs) over robust optimal point obtained by original GP surrogate for $\theta = 0.25$. Augmented parametric bootstrapping is performed using on hand set of input/output data provided among original optimization program.

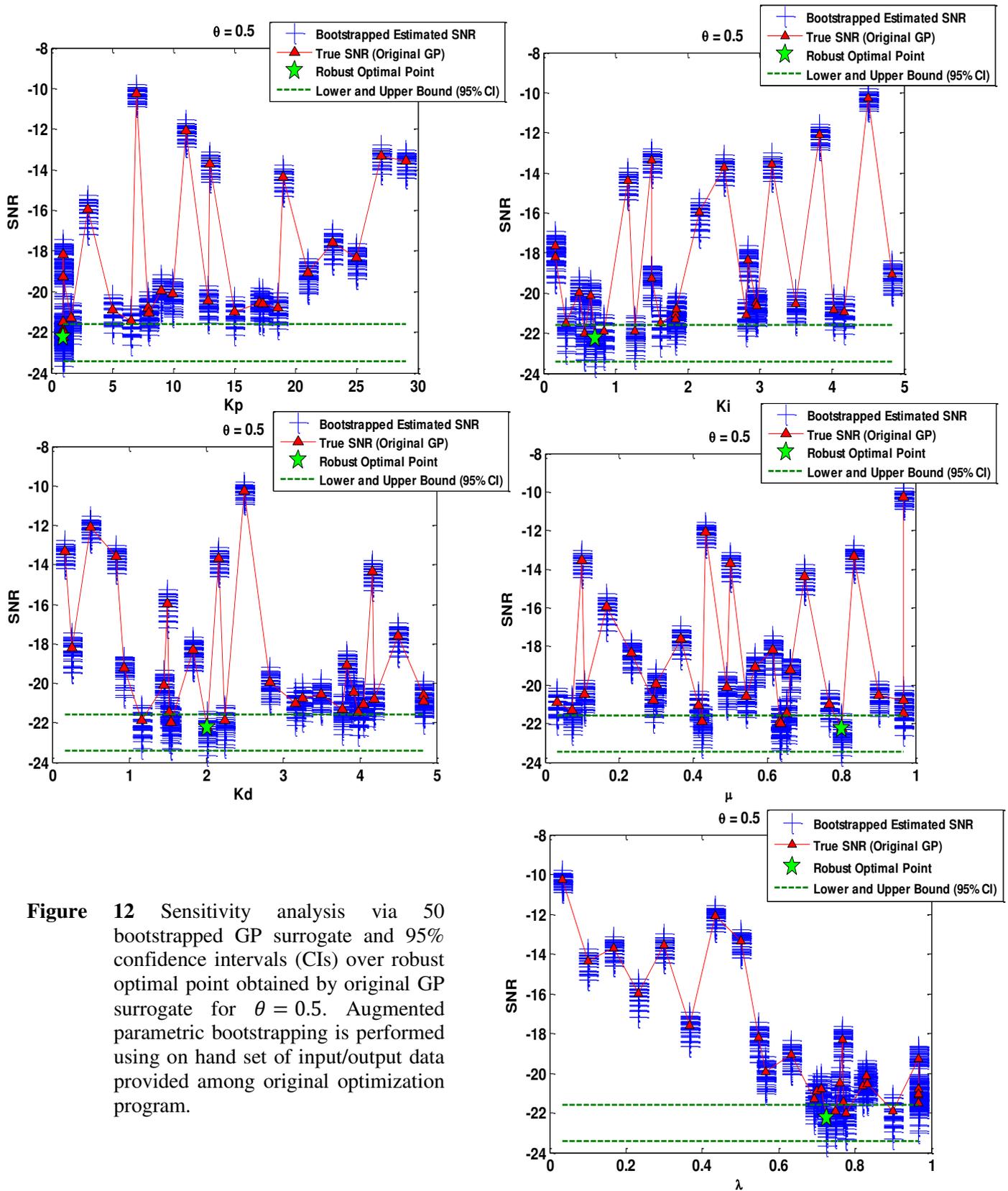


Figure 12 Sensitivity analysis via 50 bootstrapped GP surrogate and 95% confidence intervals (CIs) over robust optimal point obtained by original GP surrogate for $\theta = 0.5$. Augmented parametric bootstrapping is performed using on hand set of input/output data provided among original optimization program.

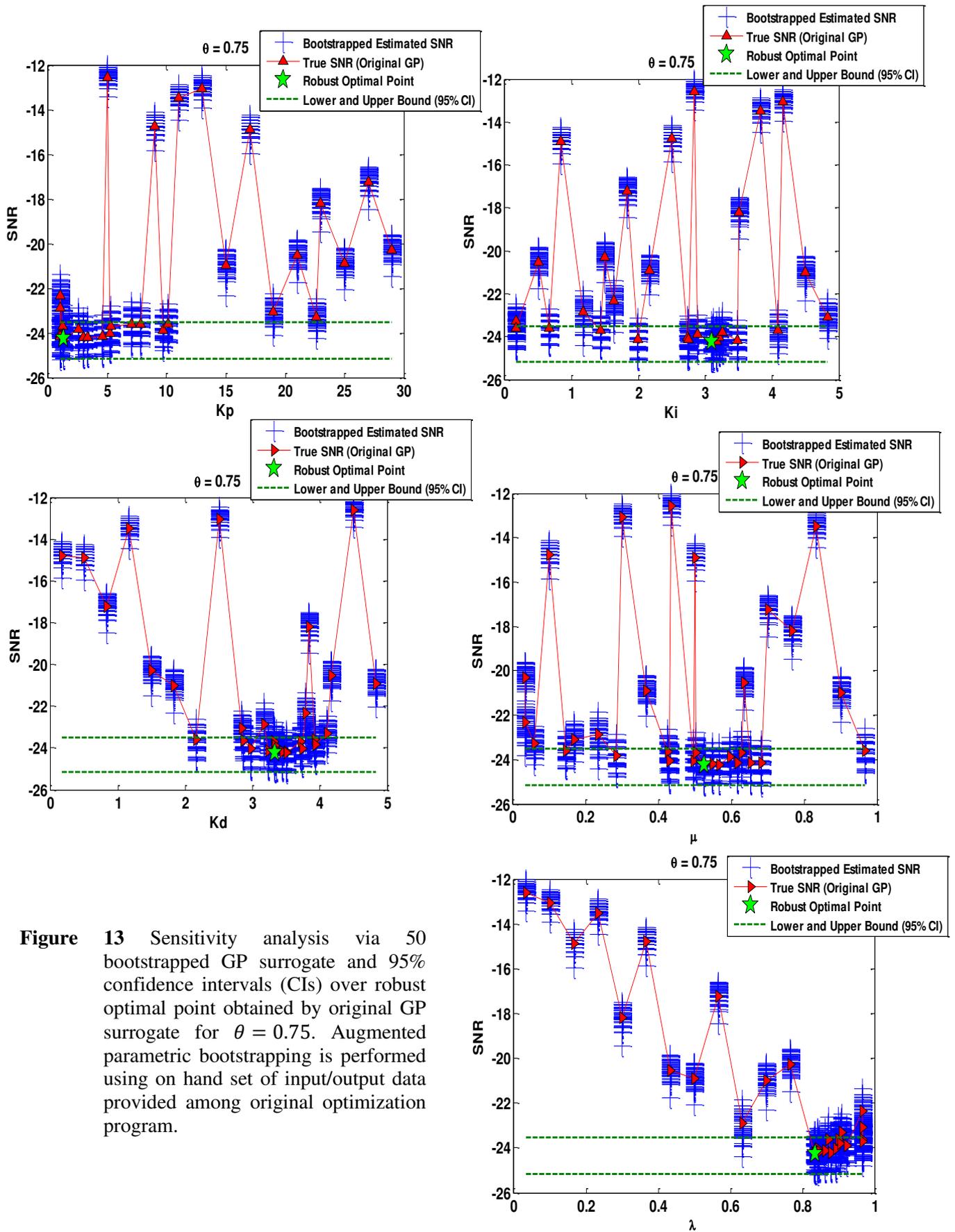


Figure 13 Sensitivity analysis via 50 bootstrapped GP surrogate and 95% confidence intervals (CIs) over robust optimal point obtained by original GP surrogate for $\theta = 0.75$. Augmented parametric bootstrapping is performed using on hand set of input/output data provided among original optimization program.

5. Conclusion

In this paper, a new hybrid surrogate/metaheuristic-based robust simulation-optimization algorithm is proposed that possesses the advantages of both GP surrogate in learning the behavior of the model in efficient global optimization and PSO metaheuristic in convergence searching of optimum results. We smooth the application of PSO using GP surrogate instead of the original simulation model to diminish computational cost due to a large number of fitness evaluations required for the global optimizer when used individually. Also, this simple modified algorithm is developed in such a way to handle computational complexity to obtain optimal and robust FOPID design in the CPS control system. In such a system, we also consider conflict between multiple objective functions and uncertainty in the model. Here, we apply this approach to the robust optimal control design of a five-bar linkage robot manipulator to depict the applicability and effectiveness of the proposed algorithm. Comparative simulation results reveal that the proposed hybrid GP/PSO-based robust efficient global optimization algorithm can effectively and robustly tune the parameters of the FOPID controllers. From an application point of view, the introduced technique is simple and fast and has a suitable control on error and energy of system and it can be easily implemented in real-world applications of CPS control systems.

Declarations

- **Acknowledgements**

The authors acknowledge the support by Second Century Fund (C2F), Chulalongkorn University, Bangkok for funding and postdoctoral fellowship.

- **Authors' contributions**

All authors have contributed to this study. All authors read and approved the final manuscript.

- **Funding**

This research project is supported by Second Century Fund (C2F), Chulalongkorn University, Bangkok.

- **Availability of data and materials**

The authors share all the materials including Matlab® codes and functions constructed through the development of the proposed algorithm and implement it for the robust optimal design of FOPID controller in google-drive folder as linked below:

<https://drive.google.com/drive/folders/1xgaT8KoLAM0zdhlQlvNy8S8bO902CG8?usp=sharing>

- **Competing interests**

The authors declare that they have no competing interests.

References

1. W. C. M. van Beers and J. P. C. Kleijnen, *Kriging for interpolation in random simulation*, Journal of the Operational Research Society, 54, (3), pp. 255–262, 2003.
2. G. Figueira and B. Almada-Lobo, *Hybrid simulation optimization methods a taxonomy and discussion*, Simulation Modelling Practice and Theory, 46, pp. 118–134, 2014.
3. S. Amaran, N. V. Sahinidis, B. Sharda, and S. J. Bury, *Simulation optimization: a review of algorithms and applications*, Annals of Operations Research, 240, (1), pp. 351–380, 2016.
4. A. Parnianifard, A. S. Azfanizam, M. K. A. Ariffin, and M. I. S. Ismail, *An overview on robust design hybrid metamodeling : Advanced methodology in process optimization under uncertainty*, International Journal of Industrial Engineering Computations, 9, (1), pp. 1–32, 2018.
5. A. Parnianifard, A. Azfanizam, M. Ariffin, M. Ismail, and N. Ebrahim, *Recent developments in metamodel based robust black-box simulation optimization: An overview*, Decision Science Letters, 8, (1), pp. 17–44, 2019.
6. R. Skowroński, *The open blockchain-aided multi-agent symbiotic cyber-physical systems*, Future Generation Computer Systems, 94, pp. 430–443, 2019.
7. E. A. Lee, *The past, present and future of cyber-physical systems: A focus on models*, Sensors (Switzerland), 15, (3), pp. 4837–4869, 2015.
8. E. A. Lee, *Cyber physical systems: Design challenges*, in *Proceedings - 11th IEEE Symposium on Object/Component/Service-Oriented Real-Time Distributed Computing, ISORC 2008*, 2008, pp. 363–369.
9. E. a Lee, *Computing Foundations and Practice for Cyber- Physical Systems : A Preliminary Report*, Electrical Engineering, (UCB/EECS-2007-72), pp. 1–27, 2007.

10. M. Zamani, *Control of cyber-physical systems using incremental properties of physical systems*, 2012.
11. C. Koulamas and A. Kalogeras, *Cyber-physical systems and digital twins in the industrial internet of things*, *Computer*, 51, (11), pp. 95–98, 2018.
12. F. Tao, Q. Qi, L. Wang, and A. Y. C. Nee, *Digital Twins and Cyber-Physical Systems toward Smart Manufacturing and Industry 4.0: Correlation and Comparison*, *Engineering*, 5, (4), pp. 653–661, 2019.
13. L. Hu, N. Xie, Z. Kuang, and K. Zhao, *Review of cyber-physical system architecture*, *Proceedings - 2012 15th IEEE International Symposium on Object/Component/Service-Oriented Real-Time Distributed Computing Workshops, ISORCW 2012*, pp. 25–30, 2012.
14. K. Sampigethaya and R. Poovendran, *Aviation cyber-physical systems: Foundations for future aircraft and air transport*, *Proceedings of the IEEE*, 101, (8), pp. 1834–1855, 2013.
15. I. S. Sacala, M. A. Moisescu, and D. Repta, *Towards the development of the future internet based enterprise in the context of cyber-physical systems*, in *2013 19th International Conference on Control Systems and Computer Science*, 2013, pp. 405–412.
16. K. Sampigethaya and R. Poovendran, *Cyber-physical integration in future aviation information systems*, in *2012 IEEE/AIAA 31st Digital Avionics Systems Conference (DASC)*, 2012, pp. 7C2-1.
17. A. Banerjee, K. K. Venkatasubramanian, T. Mukherjee, and S. K. S. Gupta, *Ensuring safety, security, and sustainability of mission-critical cyber-physical systems*, *Proceedings of the IEEE*, 100, (1), pp. 283–299, 2011.
18. C. W. Axelrod, *Managing the risks of cyber-physical systems*, in *2013 IEEE Long Island Systems, Applications and Technology Conference (LISAT)*, 2013, pp. 1–6.
19. M. J. Stanovich *et al.*, *Development of a smart-grid cyber-physical systems testbed*, in *2013 IEEE PES Innovative Smart Grid Technologies Conference (ISGT)*, 2013, pp. 1–6.
20. J. Taneja, R. Katz, and D. Culler, *Defining cps challenges in a sustainable electricity grid*, in *2012 IEEE/ACM Third International Conference on Cyber-Physical Systems*, 2012, pp. 119–128.
21. M. Ghorbani and P. Bogdan, *A cyber-physical system approach to artificial pancreas design*, 2013 International Conference on Hardware/Software Codesign and System Synthesis, CODES+ISSS 2013, pp. 1–10, 2013.
22. H. Wang, X. Deng, and F. Tian, *WiP abstract: A human-centered cyber-physical systematic approach for post-stroke monitoring*, in *2012 IEEE/ACM Third International Conference on Cyber-Physical Systems*, 2012, p. 209.
23. A. Banerjee and S. K. S. Gupta, *Spatio-temporal hybrid automata for safe cyber-physical systems: A medical case study*, in *2013 ACM/IEEE International Conference on Cyber-Physical Systems (ICCPS)*, 2013, pp. 71–80.
24. C. Sankavaram, A. Kodali, and K. Pattipati, *An integrated health management process for automotive cyber-physical systems*, in *2013 International Conference on Computing, Networking and Communications (ICNC)*, 2013, pp. 82–86.
25. Y. P. Fallah and R. Sengupta, *A cyber-physical systems approach to the design of vehicle safety networks*, in *2012 32nd International Conference on Distributed Computing Systems Workshops*, 2012, pp. 324–329.
26. X. Li *et al.*, *A holistic approach to service delivery in driver-in-the-loop vehicular CPS*, *IEEE Journal on Selected Areas in Communications*, 31, (9), pp. 513–522, 2013.

27. M. Lukaszewycz *et al.*, *Cyber-physical systems design for electric vehicles*, in *2012 15th Euromicro Conference on Digital System Design*, 2012, pp. 477–484.
28. G. Schirner, D. Erdogmus, K. Chowdhury, and T. Padir, *The future of human-in-the-loop cyber-physical systems*, *Computer*, 46, (1), pp. 36–45, 2013.
29. M. Franke, C. Seidl, and T. Schlegel, *A seamless integration, semantic middleware for cyber-physical systems*, in *2013 10th IEEE INTERNATIONAL CONFERENCE ON NETWORKING, SENSING AND CONTROL (ICNSC)*, 2013, pp. 627–632.
30. S. El-Tawab and S. Olariu, *Communication protocols in FRIEND: A cyber-physical system for traffic Flow Related Information Aggregation and Dissemination*, in *2013 IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*, 2013, pp. 447–452.
31. A. Aminifar, P. Eles, Z. Peng, and A. Cervin, *Control-quality driven design of cyber-physical systems with robustness guarantees*, in *2013 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2013, pp. 1093–1098.
32. Z. Feng, J. Wang, Y. Ma, and Y. Tu, *Robust parameter design based on Gaussian process with model uncertainty*, *International Journal of Production Research*, 0, (0), pp. 1–17, 2020.
33. F. Hu *et al.*, *Robust Cyber-Physical Systems: Concept, models, and implementation*, *Future Generation Computer Systems*, 56, pp. 449–475, 2016.
34. Q. Zhu, C. Rieger, and T. Başar, *A hierarchical security architecture for cyber-physical systems*, *Proceedings - ISRCS 2011: 4th International Symposium on Resilient Control Systems*, pp. 15–20, 2011.
35. B. Ali Asghar, *Computational Intelligence and Its Applications in Uncertainty-Based Design Optimization*, in *Bridge Optimization-Inspection and Condition Monitoring*, IntechOpen, 2019.
36. G. Wang and S. Shan, *Review of Metamodeling Techniques in Support of Engineering Design Optimization*, *Journal of Mechanical Design*, 129, (4), pp. 370–380, 2007.
37. A. Parnianifard, A. S. Azfanizam, M. K. A. Ariffin, and M. I. S. Ismail, *Comparative study of metamodeling and sampling design for expensive and semi-expensive simulation models under uncertainty*, *SIMULATION*, 96, (1), pp. 89–110, May 2019.
38. A. Parnianifard and A. . Azfanizam, *Metamodel-based robust simulation-optimization assisted optimal design of multiloop integer and fractional-order PID controller*, *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields*, 33, (1), p. e2679, 2020.
39. A. Parnianifard, A. S. Azfanizam, M. K. A. Ariffin, and M. I. S. Ismail, *Kriging-Assisted Robust Black-Box Simulation Optimization in Direct Speed Control of DC Motor Under Uncertainty*, *IEEE Transactions on Magnetics*, 54, (7), pp. 1–10, 2018.
40. T. W. Simpson, J. D. Poplinski, P. N. Koch, and J. K. Allen, *Metamodels for Computer-based Engineering Design: Survey and recommendations*, *Engineering With Computers*, 17, (2), pp. 129–150, 2001.
41. Y. F. Li, S. H. Ng, M. Xie, and T. N. Goh, *A systematic comparison of metamodeling techniques for simulation optimization in Decision Support Systems*, *Applied Soft Computing*, 10, (4), pp. 1257–1273, 2010.
42. R. Jin, X. Du, and W. Chen, *The use of metamodeling techniques for optimization under uncertainty*, 25, (2). 2003.
43. R. Yondo, E. Andrés, and E. Valero, *A review on design of experiments and surrogate*

models in aircraft real-time and many-query aerodynamic analyses, Progress in Aerospace Sciences, 96, pp. 23–61, 2018.

44. R. Rajkumar, I. Lee, L. Sha, and J. Stankovic, *Cyber-physical systems: The next computing revolution*, Proceedings - Design Automation Conference, pp. 731–736, 2010.
45. P. Shah and S. Agashe, *Review of fractional PID controller*, Mechatronics, 38, (January 2020), pp. 29–41, 2016.
46. R. Ranganayakulu, G. Uday Bhaskar Babu, A. Seshagiri Rao, and D. S. Patle, *A comparative study of fractional order PI λ /PI λ D μ tuning rules for stable first order plus time delay processes*, Resource-Efficient Technologies, 2, pp. S136–S152, 2016.
47. A. Tepljakov, B. B. Alagoz, C. Yeroglu, E. Gonzalez, S. H. HosseinNia, and E. Petlenkov, *FOPID Controllers and Their Industrial Applications: A Survey of Recent Results I*, IFAC-PapersOnLine, 51, (4), pp. 25–30, 2018.
48. I. Podlubny, *Fractional-order systems and fractional-order controllers*, Institute of Experimental Physics, Slovak Academy of Sciences, Kosice, 12, (3), pp. 1–18, 1994.
49. M. A. Clark and K. S. Rattan, *Piecewise affine hybrid automata representation of a multistage fuzzy PID controller*, AAI Spring Symposium - Technical Report, SS-14-02, pp. 104–109, 2014.
50. W. W. Shein, Y. Tan, and A. O. Lim, *PID controller for temperature control with multiple actuators in cyber-physical home system*, in *2012 15th International Conference on Network-Based Information Systems*, 2012, pp. 423–428.
51. W. Wang, F. Di Maio, and E. Zio, *Hybrid fuzzy-PID control of a nuclear Cyber-Physical System working under varying environmental conditions*, Nuclear Engineering and Design, 331, (December 2017), pp. 54–67, 2018.
52. K. Miettinen, *Nonlinear multiobjective optimization*, 12. Springer Science & Business Media, 2012.
53. K. M. Miettinen, *Nonlinear multiobjective optimization*, 12. Springer Science & Business Media, 1998.
54. C. E. Rasmussen and C. K. I. Williams, *Gaussian processes for machine learning*. 2006, 38, (2). 2006.
55. J. P. C. Kleijnen, *Kriging metamodeling in simulation: A review*, European Journal of Operational Research, 192, (3), pp. 707–716, 2009.
56. J. P. C. C. Kleijnen, *Design and analysis of simulation experiments (2nd)*. Springer, 2017.
57. T. W. Simpson, T. M. Mauery, J. Korte, and F. Mistree, *Kriging models for global approximation in simulation-based multidisciplinary design optimization*, AIAA Journal, 39, (12), pp. 2233–2241, 2001.
58. A. Parnianifard, A. S. Azfanizam, M. K. Ariffin, M. I. Ismail, M. R. Maghami, and C. Gomes, *Kriging and Latin Hypercube Sampling Assisted Simulation Optimization in Optimal Design of PID Controller for Speed Control of DC Motor*, Journal of Computational and Theoretical Nanoscience, 15, (5), pp. 1471–1479, 2018.
59. R. Eberhart and J. Kennedy, *Particle swarm optimization*, in *Proceedings of the IEEE international conference on neural networks*, 1995, 4, pp. 1942–1948.
60. M. N. Ab Wahab, S. Nefti-Meziani, and A. Atyabi, *A Comprehensive Review of Swarm Optimization Algorithms*, PLOS ONE, 10, (5), p. e0122827, May 2015.

61. Y. Del Valle, G. K. Venayagamoorthy, S. Mohagheghi, J.-C. Hernandez, and R. G. Harley, *Particle swarm optimization: basic concepts, variants and applications in power systems*, IEEE Transactions on evolutionary computation, 12, (2), pp. 171–195, 2008.
62. M. P. Aghababa, *Optimal design of fractional-order PID controller for five bar linkage robot using a new particle swarm optimization algorithm*, Soft Computing, 20, (10), pp. 4055–4067, 2016.
63. M. Zamani, M. Karimi-Ghartemani, N. Sadati, and M. Parniani, *Design of a fractional order PID controller for an AVR using particle swarm optimization*, Control Engineering Practice, 17, (12), pp. 1380–1387, 2009.
64. H. Yu, Y. Tan, J. Zeng, C. Sun, and Y. Jin, *Surrogate-assisted hierarchical particle swarm optimization*, Information Sciences, 454–455, pp. 59–72, 2018.
65. S. Dutta, *A sequential metamodel-based method for structural optimization under uncertainty*, Structures, 26, (July 2019), pp. 54–65, 2020.
66. R. G. Regis, *Particle swarm with radial basis function surrogates for expensive black-box optimization*, Journal of Computational Science, 5, (1), pp. 12–23, 2014.
67. G. Dellino, P. C. Kleijnen, Jack, and C. Meloni, *Metamodel-Based Robust Simulation-Optimization: An Overview*, in *In Uncertainty Management in Simulation-Optimization of Complex Systems*, Springer US, 2015, pp. 27–54.
68. A. Parnianifard, A. S. Azfanizam, M. K. A. Ariffin, and M. I. S. Ismail, *Crossing weighted uncertainty scenarios assisted distribution-free metamodel-based robust simulation optimization*, Engineering with Computers, 36, (1), pp. 139–150, 2019.
69. S. Park and J. Antony, *Robust design for quality engineering and six sigma*. World Scientific Publishing Co Inc, 2008.
70. M. S. Phadke, *Quality Engineering Using Robust Design*. Prentice Hall PTR, 1989.
71. F. Jurecka, *Robust Design Optimization Based on Metamodeling Techniques*, PhD Thesis, 2007.
72. J. Havinga, A. H. van den Boogaard, and G. Klaseboer, *Sequential improvement for robust optimization using an uncertainty measure for radial basis functions*, Structural and Multidisciplinary Optimization, 55, (4), pp. 1345–1363, 2017.
73. N. Drira *et al.*, *Convergence rates of the efficient global optimization algorithm for improving the design of analog circuits*, Analog Integrated Circuits and Signal Processing, 103, (1), pp. 143–162, 2020.
74. K. Rutten, *Methods For Online Sequential Process Improvement*, PhD Thesis, 2015.
75. J. P. C. Kleijnen, W. Van Beers, and I. Van Nieuwenhuyse, *Expected improvement in efficient global optimization through bootstrapped kriging*, Journal of Global Optimization, 54, (1), pp. 59–73, 2012.
76. M. H. Quenouille, *Approximate tests of correlation in time-series 3*, in *Mathematical Proceedings of the Cambridge Philosophical Society*, 1949, 45, (3), pp. 483–484.
77. J. Tukey, *Bias and confidence in not quite large samples*, Ann. Math. Statist., 29, p. 614, 1958.
78. R. Nisbet, J. Elder, and G. Miner, *Handbook of statistical analysis and data mining-2nd*. Academic Press., 2017.
79. M. D. McKay, R. J. Beckman, and W. J. Conover, *Comparison of three methods for*

selecting values of input variables in the analysis of output from a computer code, Technometrics, 21, (2), pp. 239–245, 1979.

80. R. L. Iman and W. J. Conover, *A distribution-free approach to inducing rank correlation among input variab*, *Communications in Statistics - Simulation and Computation*, 11, (3), pp. 311–334, 1982.
81. F. A. C. Viana, *A Tutorial on Latin Hypercube Design of Experiments*, *Quality and Reliability Engineering International*, 32, (5), pp. 1975–1985, 2016.
82. R. C. . Cheng, *Resampling methods*, *Handbooks in operations research and management science*, 13, pp. 415–453, 2006.
83. G. Dellino, J. P. C. Kleijnen, and C. Meloni, *Robust optimization in simulation: Taguchi and Krige combined*, *INFORMS Journal on Computing*, 24, (3), pp. 471–484, 2012.
84. J. P. C. Kleijnen and W. C. M. van Beers, *Monotonicity-preserving bootstrapped Kriging metamodels for expensive simulations*, *Journal of the Operational Research Society*, 64, (5), pp. 708–717, 2013.
85. A. T. Azar, J. Kumar, V. Kumar, and K. P. S. Rana, *Control of a two link planar electrically-driven rigid robotic manipulator using fractional order SOFC*, in *International Conference on Advanced Intelligent Systems and Informatics*, 2017, pp. 57–68.
86. T. Kathuria, V. Kumar, K. P. S. Rana, and A. T. Azar, *Control of a Three-Link Manipulator Using Fractional-Order PID Controller*, in *Fractional Order Systems*, Elsevier Inc., 2018, pp. 477–510.
87. G. Krishan and V. R. Singh, *Motion control of five bar linkage manipulator using conventional controllers under uncertain conditions*, *International Journal of Intelligent Systems and Applications*, 8, (5), pp. 34–40, 2016.
88. M. W. Spong, S. Hutchinson, and M. Vidyasagar, *Robot modeling and control*, (Apr 13). 2020.
89. A. Tepljakov, E. Petlenkov, and J. Belikov, *FOMCON : a MATLAB Toolbox for Fractional-order System Identification and Control*, *International Journal of Microelectronics and Computer Science*, 2, (2), pp. 51–62, 2011.
90. M. A. Badamchizadeh, I. Hassanzadeh, and M. Abedinpour Fallah, *Extended and unscented kalman filtering applied to a flexible-joint robot with jerk estimation*, *Discrete Dynamics in Nature and Society*, 2010, 2010.
91. S. N. Lophaven, H. B. Nielsen, J. Søndergaard, and H. B. Nielsen, *DACE — A Matlab Kriging Toolbox (Version 2.0)*, *IMM Informatics and Mathematical Modelling*, pp. 1–34, 2002.
92. S. K. Verma, S. Yadav, and S. K. Nagar, *Optimization of Fractional Order PID Controller Using Grey Wolf Optimizer*, *Journal of Control, Automation and Electrical Systems*, 28, (3), pp. 314–322, 2017.
93. R. Pradhan, S. K. Majhi, J. K. Pradhan, and B. B. Pati, *Optimal fractional order PID controller design using Ant Lion Optimizer*, *Ain Shams Engineering Journal*, (xxxx), 2019.
94. S. Mirjalili, S. M. Mirjalili, and A. Lewis, *Grey Wolf Optimizer*, *Advances in Engineering Software*, 69, pp. 46–61, 2014.
95. S. Mirjalili, *The ant lion optimizer*, *Advances in Engineering Software*, 83, pp. 80–98, 2015.

Figures

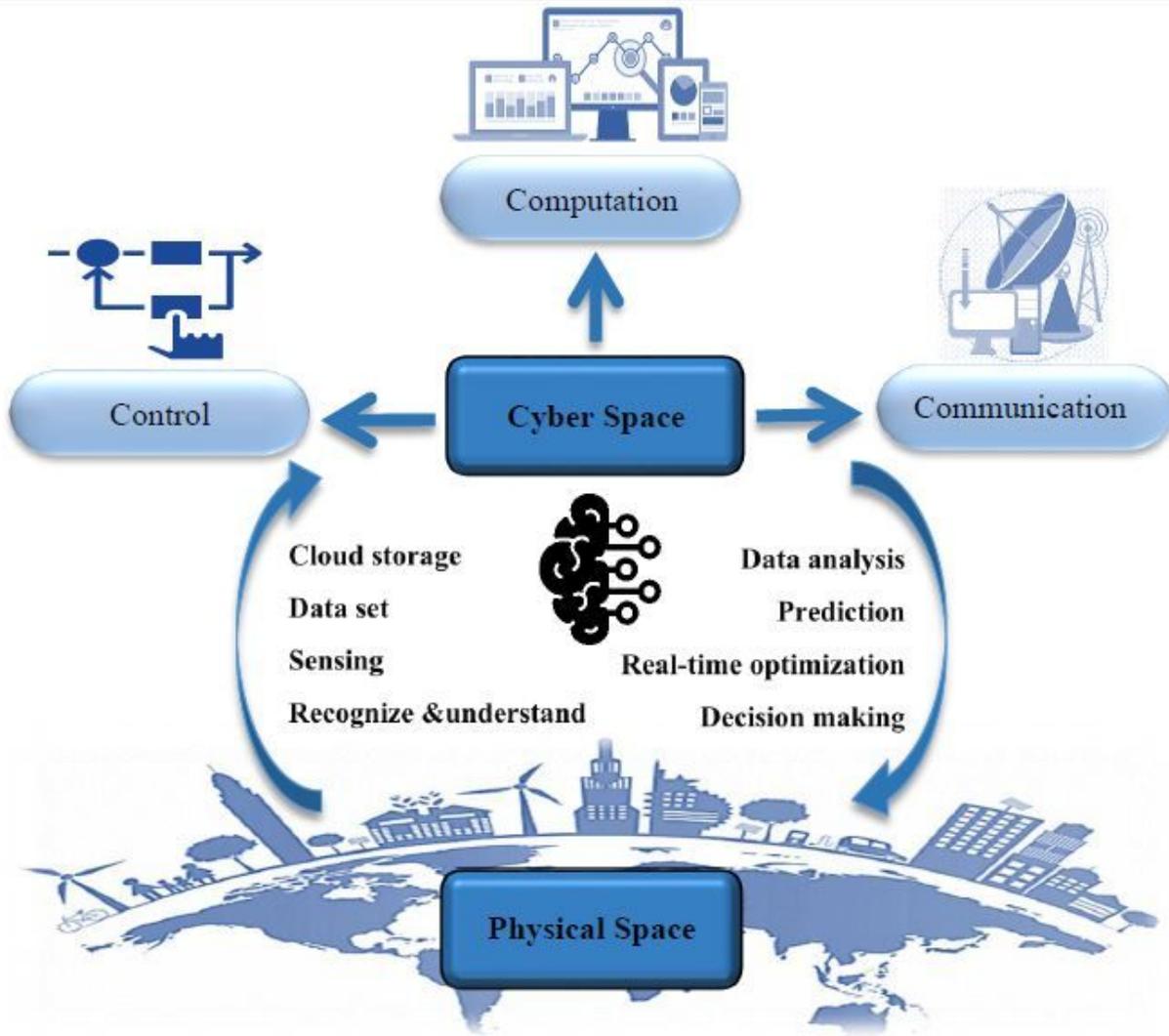


Figure 1

Representation of cyber physical system (CPS).

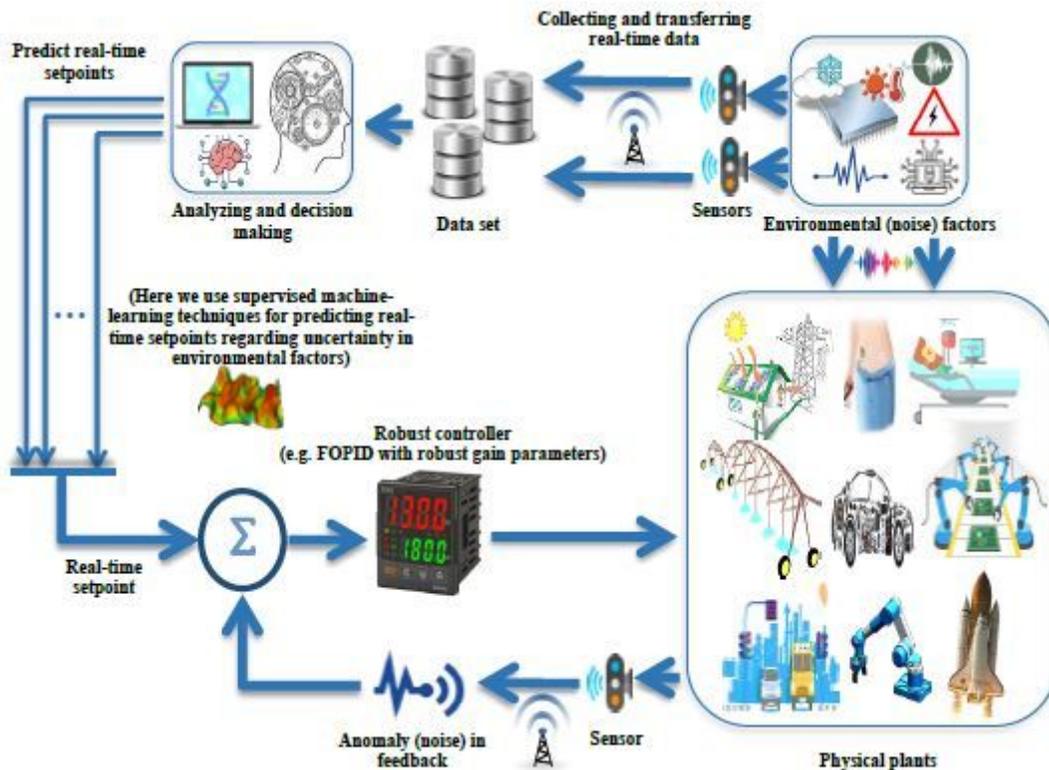


Figure 2

The control framework of CPS with real time setpoints and noise in model's feedback. The environmental factors would be predicted and applied as a real time setpoint and anomaly in sensor is estimated in feedback loop. Gain parameters and order parameters in FOPID controller are tuned to be robust against source of variability.

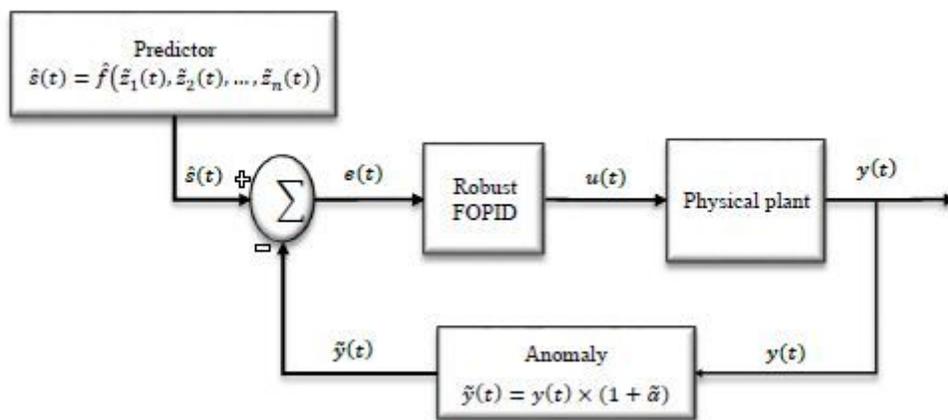


Figure 3

The block diagram of robust FOPID control in CPS framework with real time setpoints and noise in model's feedback. Real time setpoint is estimated by approximation function of environmental factors ($\hat{z}_1(t), \hat{z}_2(t), \dots, \hat{z}_n(t)$). Anomaly in sensor's feedback is function of uncertain variable \tilde{a} . FOPID

gain parameters and order parameters are tuned robustly in such a way to make CPS insensitive against sources of variability in system.

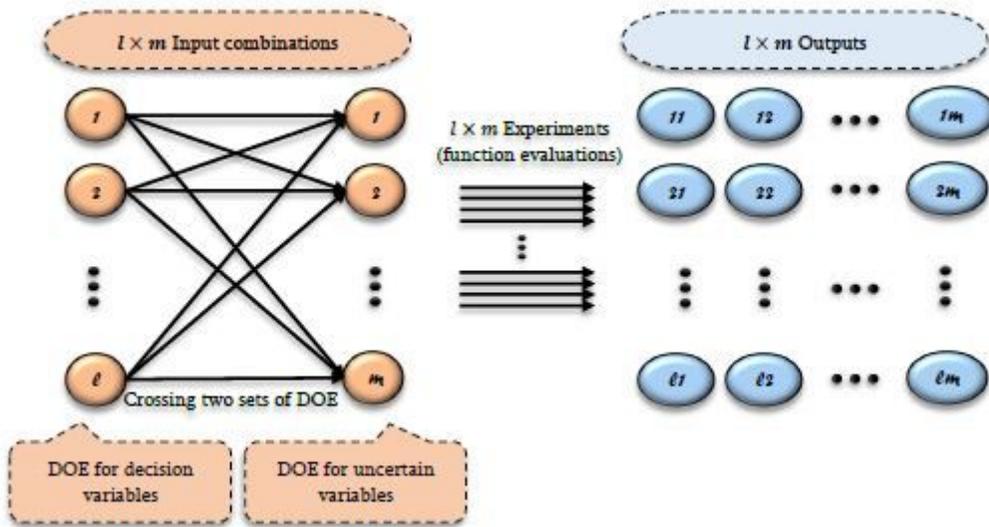


Figure 4

Crossing two sets of DOE dealing with uncertainty in a model, one DOE (n samples) over decision variables of the model and second DOE (m samples) over uncertain variables in the model

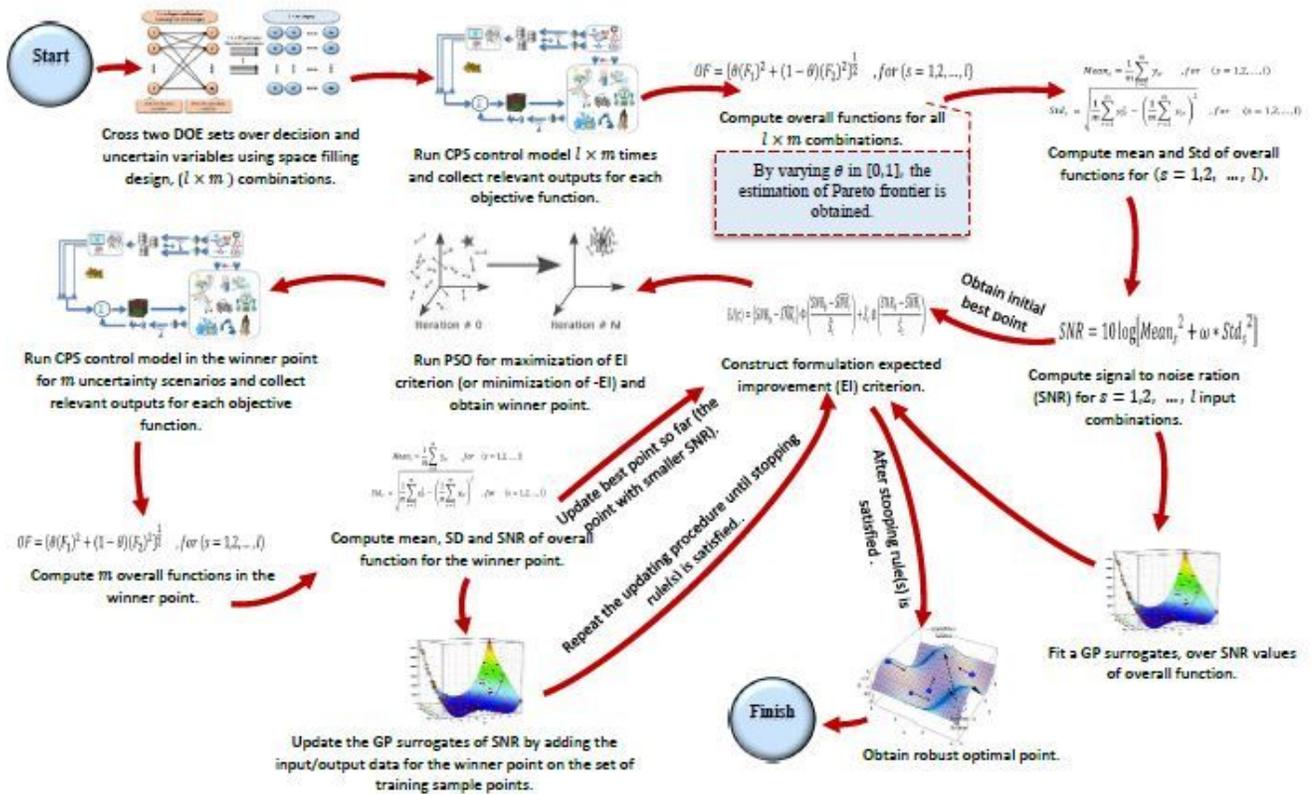


Figure 5

Algorithmic representation of proposed approach for hybrid GP PSO based robust simulation optimization under uncertainty.

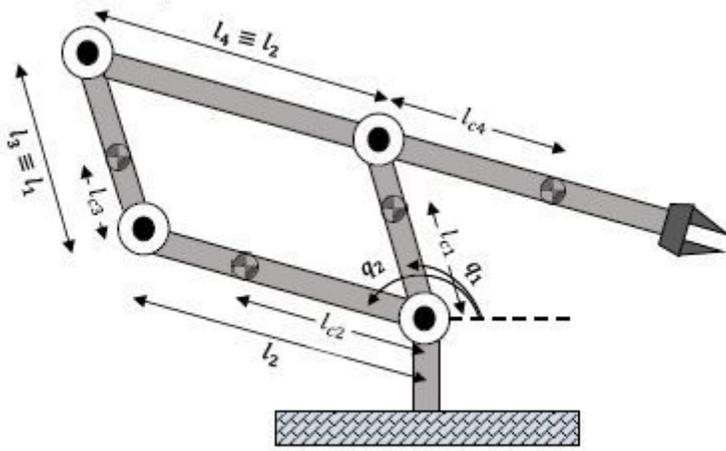


Figure 6

Five bar linkage robot manipulator.

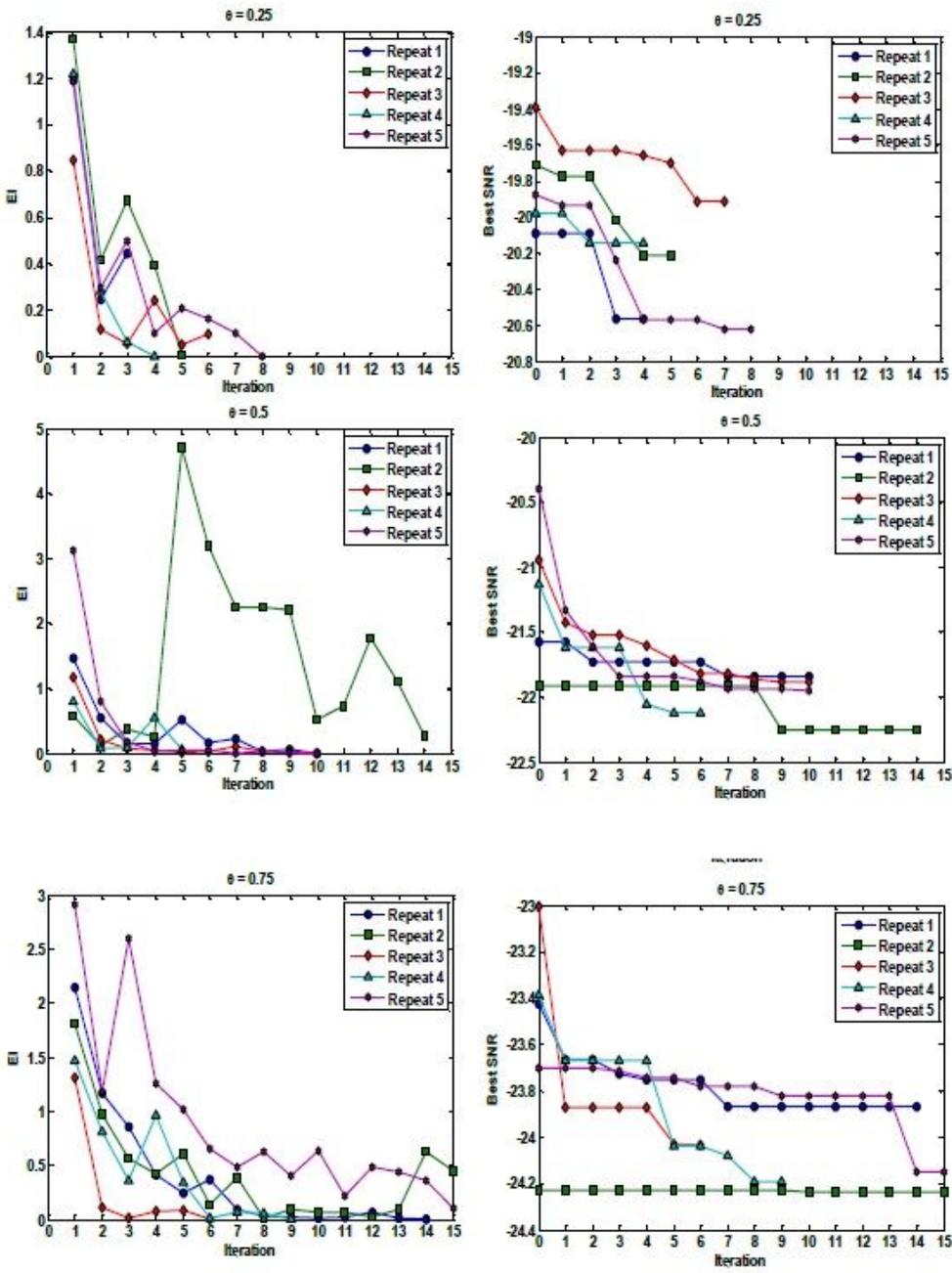


Figure 7

EI criterion magnitudes and best SNR obtained by sequential expected improvement over five different repetition of proposed algorithm for $\theta = 0.25$, $\theta = 0.5$ and $\theta = 0.75$. Two stopping rules are adjusted, EI value becomes smaller than 0.01 or reach 15 sequential iterations.

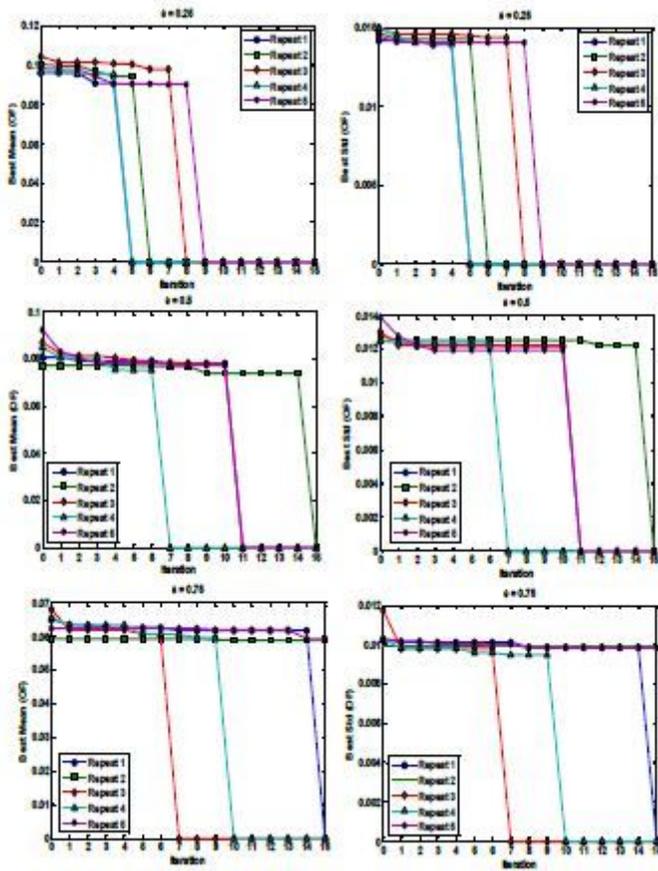


Figure 8

Mean and Std of overall function (OF) related to best point so far (smaller SNR) obtained by sequential expected improvement over five different repetition of proposed algorithm for $\alpha = 0.25$, $\alpha = 0.5$ and $\alpha = 0.75$. Two stopping rules are adjusted, EI value becomes smaller than 0.01 or reach 15 sequential iterations.

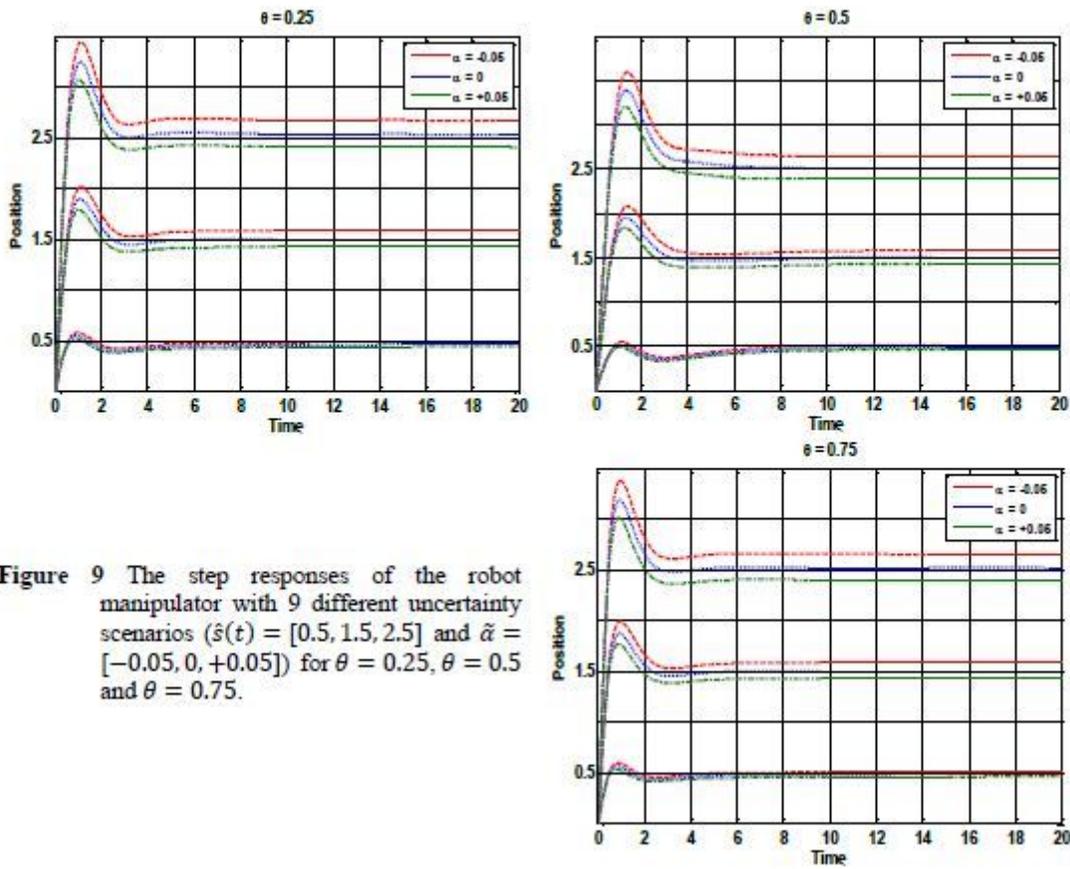


Figure 9 The step responses of the robot manipulator with 9 different uncertainty scenarios ($\hat{s}(t) = [0.5, 1.5, 2.5]$ and $\tilde{\alpha} = [-0.05, 0, +0.05]$) for $\theta = 0.25$, $\theta = 0.5$ and $\theta = 0.75$.

Figure 9

The step responses of the robot manipulator with 9 different uncertainty scenarios ($\hat{s}(t) = [0.5, 1.5, 2.5]$ and $\tilde{\alpha} = [-0.05, 0, +0.05]$) for $\theta = 0.25$, $\theta = 0.5$ and $\theta = 0.75$.

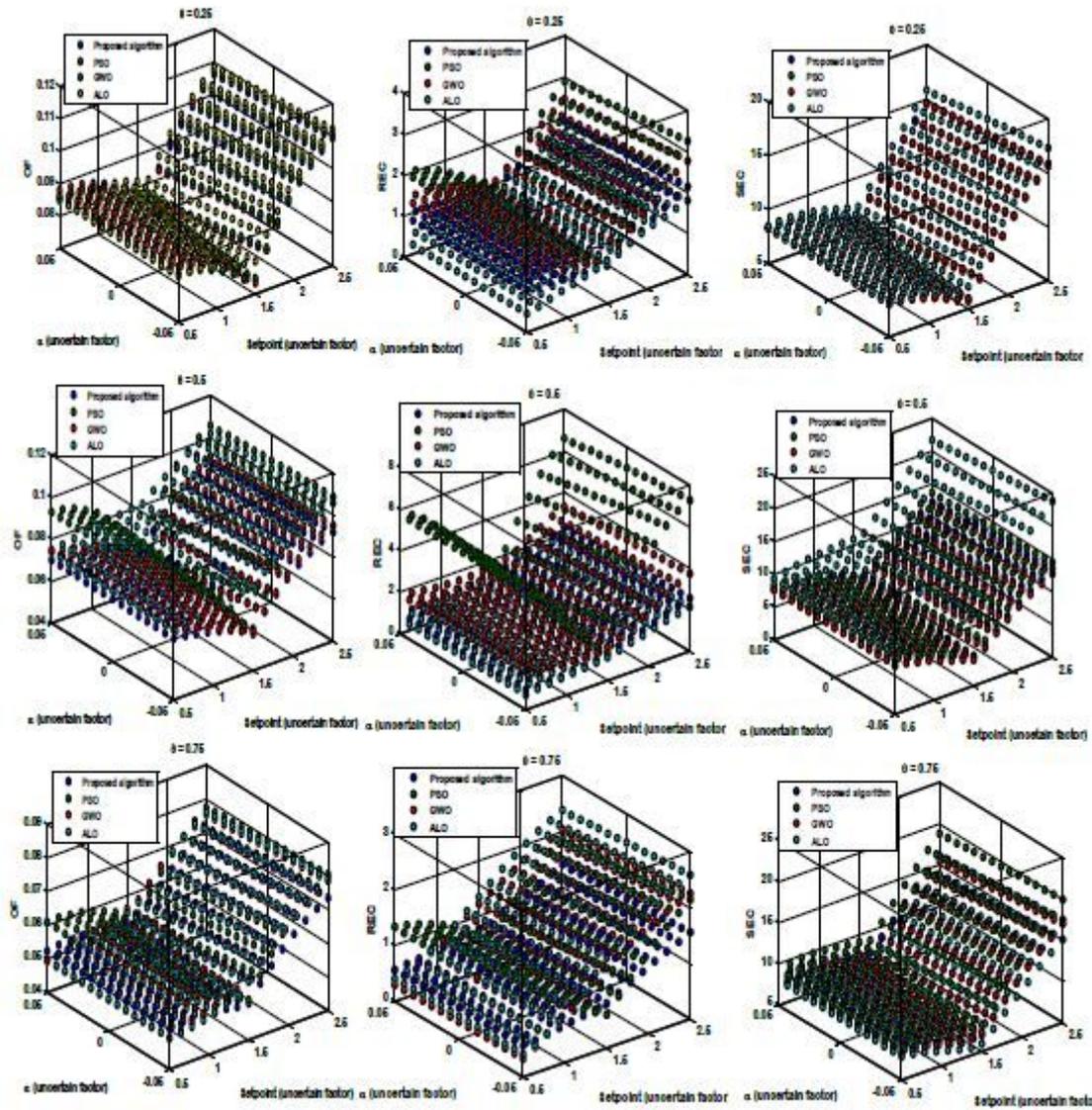


Figure 10

Comparison of proposed algorithm with three common global optimizer in obtained robust optimal point by each method for $\theta = 0.25$, $\theta = 0.5$ and $\theta = 0.75$ dealing with 200 different uncertainty scenarios.

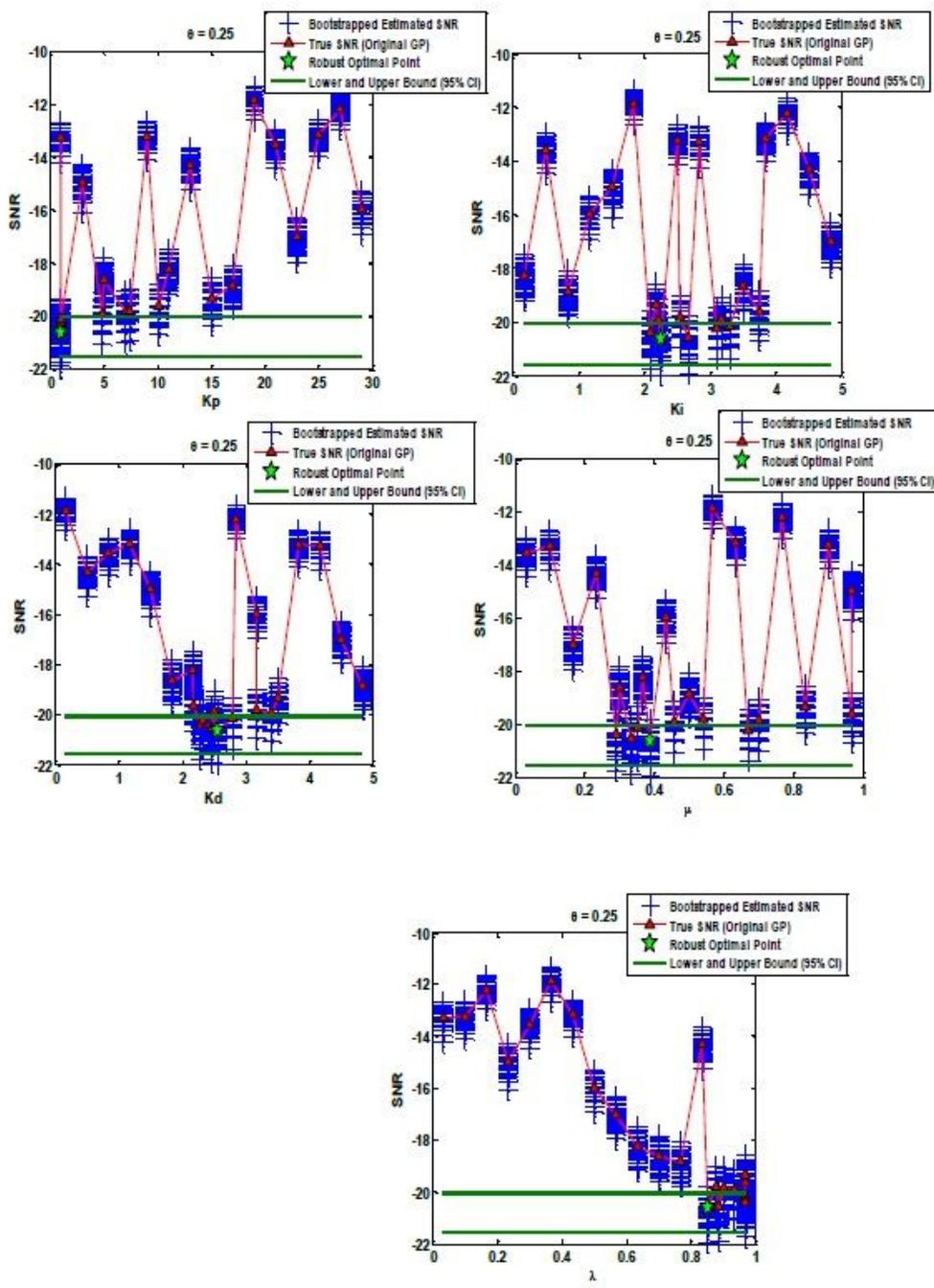


Figure 11

Sensitivity analysis via 50 bootstrapped GP surrogate and 95% confidence intervals (CIs) over robust optimal point obtained by original GP surrogate for $\theta = 0.25$. Augmented parametric bootstrapping is performed using on hand set of input/output data provided among original optimization program

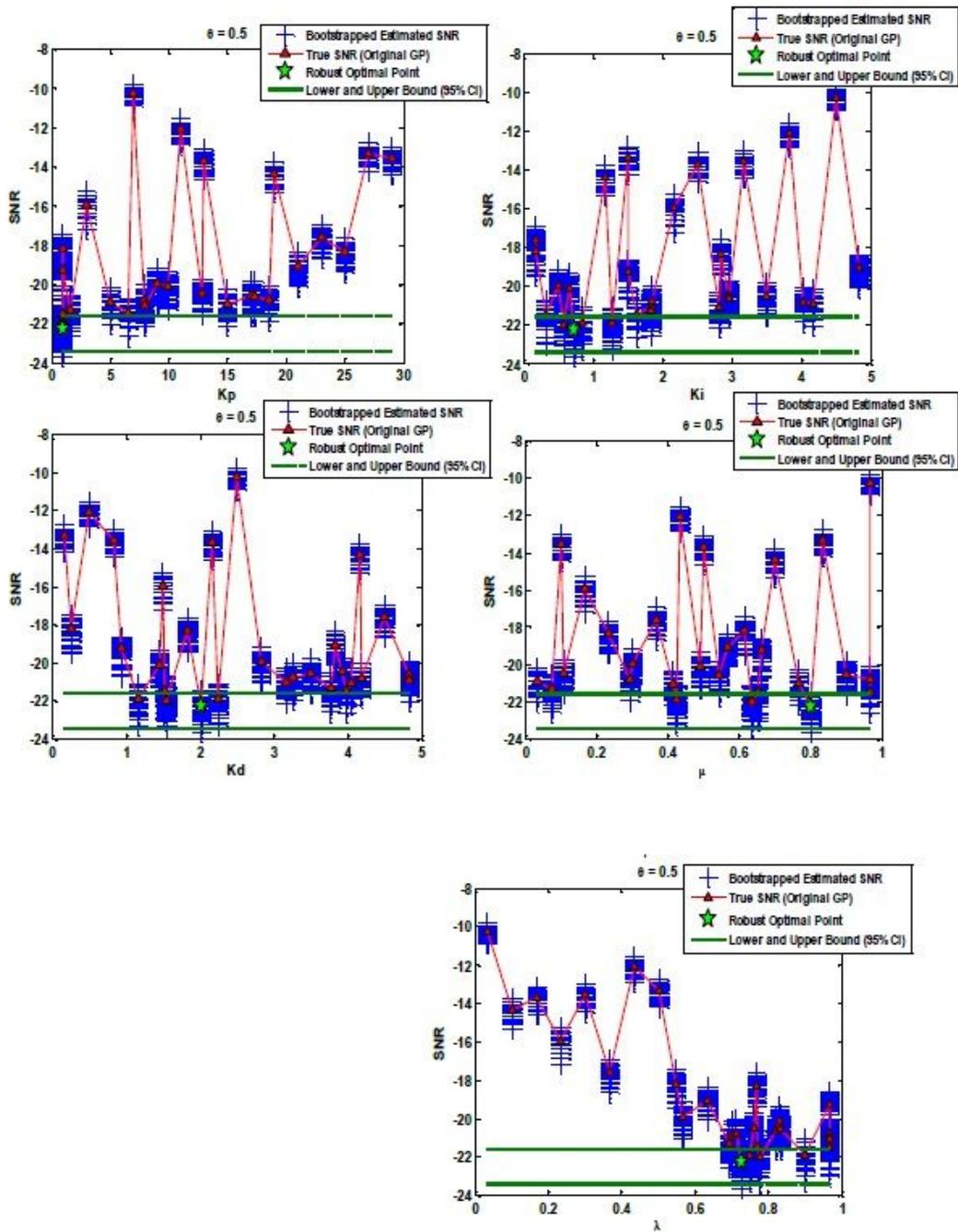


Figure 12

Sensitivity analysis via 50 bootstrapped GP surrogate and 95% confidence intervals (CIs) over robust optimal point obtained by original GP surrogate for $\theta = 0.5$. Augmented parametric bootstrapping is performed using on hand set of input/output data provided among original optimization program.

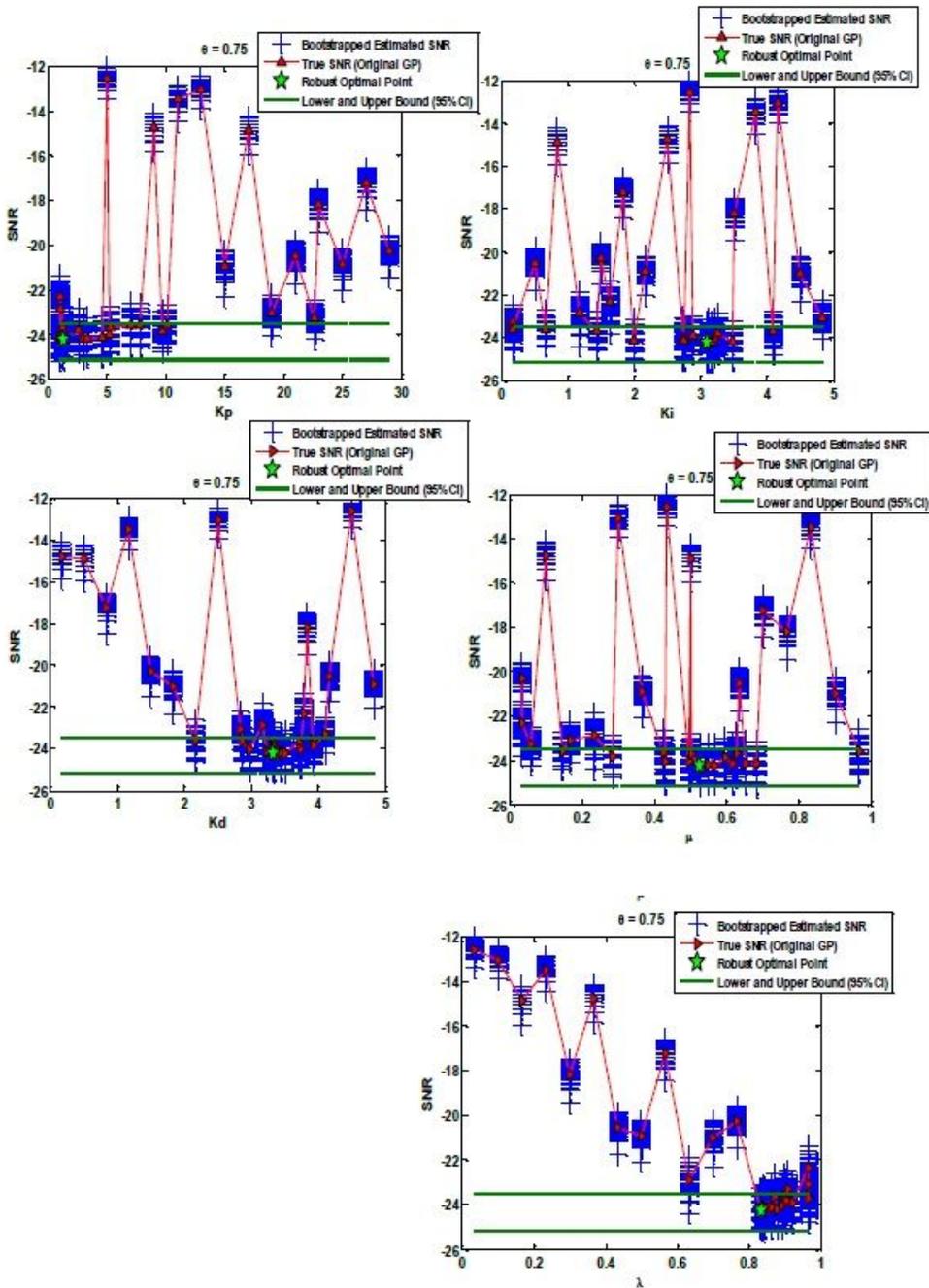


Figure 13

Sensitivity analysis via 50 bootstrapped GP surrogate and 95% confidence intervals (CIs) over robust optimal point obtained by original GP surrogate for $\theta = 0.75$. Augmented parametric bootstrapping is performed using on hand set of input/output data provided among original optimization program.